

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
Белорусский национальный технический университет

Кафедра «Робототехнические системы»

А. Р. Околов
Ю. Н. Матрунчик

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ПРОМЫШЛЕННЫХ РОБОТОВ

Учебно-методическое пособие
для студентов учреждений высшего образования
по специальностям 1-53 01 01 «Автоматизация технологических
процессов и производств», 1-53 01 06 «Промышленные роботы
и робототехнические комплексы»

*Рекомендовано учебно-методическим объединением Республики Беларусь
по образованию в области автоматизации технологических процессов,
производств и управления*

Минск
БНТУ
2021

УДК 621.865.8:004 (075.8)

ББК 32.816я7

О-51

Р е ц е н з е н т ы:

зав. кафедрой «Автоматизированные системы управления производством» БГАТУ, канд. техн. наук, доцент *А. Г. Сеньков*;

зав. лаб. робототехнических систем ОИПИ НАН Беларуси, канд. техн. наук, доцент *Г. А. Прокопович*

Околов, А. Р.

О-51 Программное обеспечение промышленных роботов: учебно-методическое пособие для студентов учреждений высшего образования по специальностям 1-53 01 01 «Автоматизация технологических процессов и производств», 1-53 01 06 «Промышленные роботы и робототехнические комплексы» / А. Р. Околов, Ю. Н. Матрунчик. – Минск : БНТУ, 2021. – 66 с.

ISBN 978-985-583-297-4.

Цель данного пособия – обеспечить методическую поддержку для приобретения студентами практических навыков составления и выполнения программ для промышленных роботов на одном из роботоориентированных языков.

Методическое пособие содержит 8 лабораторных работ, каждая работа сопровождается рекомендациями и теоретическими сведениями из области программирования промышленных роботов.

УДК 621.865.8:004 (075.8)

ББК 32.816я7

ISBN 978-985-583-297-4

© Околов А. Р., Матрунчик Ю. Н., 2021

© Белорусский национальный
технический университет, 2021

СОДЕРЖАНИЕ

Введение	5
Основные правила безопасности при выполнении лабораторных работ	6
Лабораторная работа № 1. Ознакомление с органом управления промышленного робота HIWIN RA605	7
Лабораторная работа № 2. Программирование траектории движения манипулятора	21
Лабораторная работа № 3. Аналитическое программирование промышленного робота HIWIN	34
Лабораторная работа № 4. Изучение возможностей таймеров, счетчиков ПР HIWIN RA605	41
Лабораторная работа № 5. Изучение цифровых Входов-выходов ПР HIWIN RA605	45
Лабораторная работа № 6. Введение в работу с промышленным коллаборативным роботом-манипулятором UNIVERSAL ROBOTS UR3	50
Лабораторная работа № 7. Работа с рабочим органом манипулятора UR3	60
Лабораторная работа № 8. Использование команды «pattern» для укладки однотипных деталей	64
Список используемых источников	68

ВВЕДЕНИЕ

Программное обеспечение роботов представляет собой совокупность программ и связанных с ними данных, которые разрабатываются, производятся, продаются и сопровождаются для обеспечения функционирования промышленных роботов.

Роль программного обеспечения промышленных роботов лучше всего рассматривать с трех точек зрения: специалиста по эксплуатации, разработчика роботизированной системы и производителя роботов. Требования, предъявляемые к нему различными группами пользователей, зачастую противоречивы. Например, инженер, занимающийся роботизацией, должен располагать средствами разработки программного обеспечения и сервисными программами. Они позволяют ему выполнять свою работу быстро и эффективно. С другой стороны, сохранять эти средства в программном обеспечении робота, находящегося в промышленной эксплуатации, нецелесообразно. На этом этапе стремятся повысить эксплуатационные характеристики.

Промышленный робот (ПР) предназначен для производства различных изделий. Поэтому его программное обеспечение должно обеспечивать высокую производительность при стабильном качестве выполнения операций.

Кроме того, робототехническая система должна быть надежной и простой в эксплуатации. На надежность системы влияют и аппарататура, и программное обеспечение – оно призвано обнаруживать сбои в работе оборудования, но, что еще важнее, оно само должно быть надежным. Сбои программного обеспечения часто приводят к опасным ситуациям, а ликвидация их последствий может обойтись весьма дорого.

ОСНОВНЫЕ ПРАВИЛА БЕЗОПАСНОСТИ ПРИ ВЫПОЛНЕНИИ ЛАБОРАТОРНЫХ РАБОТ

Общие правила

Перед началом цикла лабораторных работ необходимо ознакомиться с правилами безопасности при эксплуатации роботов и электроустановок.

Не включать роботов и установки, на которых лабораторные работы не выполняются.

Не касаться электропроводов, электроаппаратов, клемм и не открывать шкафы.

Не включать робота без разрешения преподавателя или инженера лаборатории.

Проверить наличие заземления.

Перед началом работы

Перед включением ПР необходимо удалить из рабочей зоны манипулятора все посторонние предметы.

Опробовать робота, используя ручное управление.

Во время работы

Категорически запрещается входить в рабочую зону работающего манипулятора.

При работе в автоматическом режиме принять меры, исключающие попадание посторонних предметов в рабочую зону робота.

Запрещается разъединять и соединять гнезда и вилки кабельных разъемов под напряжением.

При первых признаках неполадок или неуправляемости следует немедленно отключить ПР – нажать большую красную кнопку на его передней панели или рубильник общего питания лаборатории.

После работы

Выключить робота, другое оборудование и отключить от сети.

Привести в порядок рабочее место.

Лабораторная работа № 1

ОЗНАКОМЛЕНИЕ С ОРГАНОМ УПРАВЛЕНИЯ ПРОМЫШЛЕННОГО РОБОТА HIWIN RA605

Цель работы

1. Изучить основные характеристики робота HIWIN RA605.
2. Изучить интерфейс пульта ручного управления робота HIWIN RA60.
3. Приобрести навыки работы с пультом робота.

Основные сведения

Промышленный робот HIWIN серии RA605

Серия RA605 представляет собой компактного и подвижного 6-осевого робота, содержащего шарнирный рычаг с возможностью захвата. Используется для точного монтажа, комплексной обработки, манипулирования заготовкой, а также для выполнения таких функций, как подъем/перемещение, обработка, монтаж, удаление заусенцев, шлифовка и полировка.

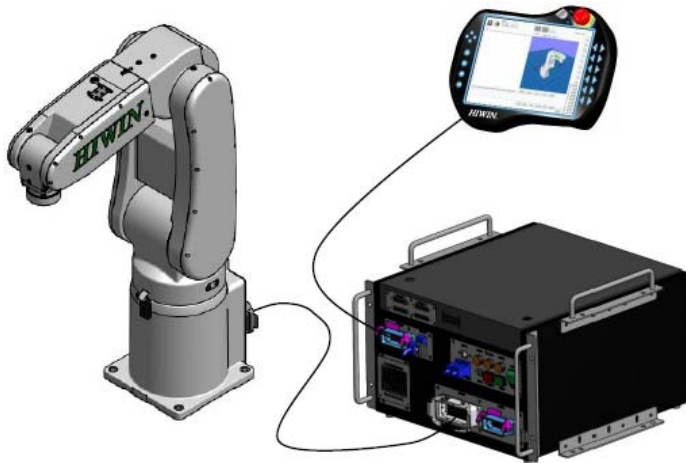
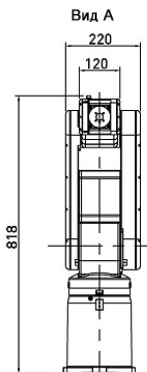
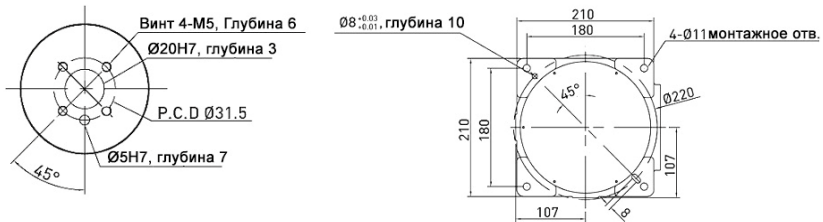
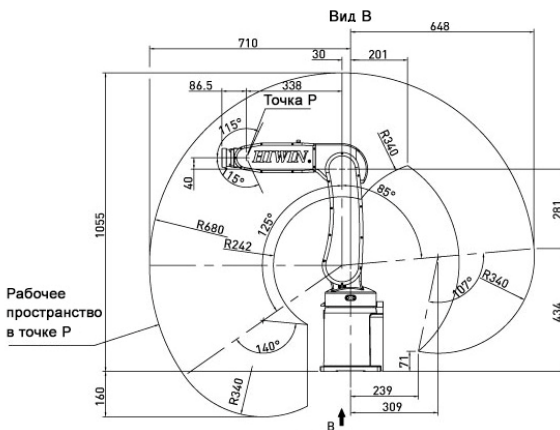


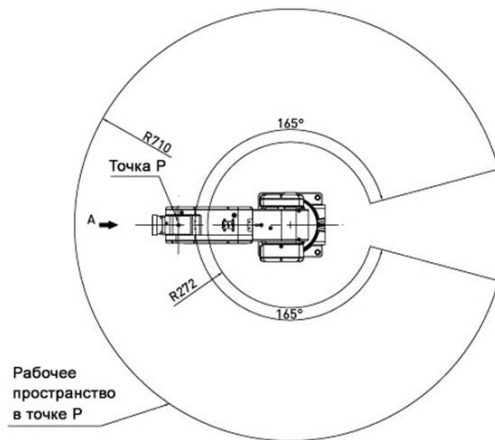
Рис. 1.1. Общий вид промышленного робота HIWIN серии RA605



Фронтальный вид



Вид сбоку



Вид сверху

Рис. 1.2. Габаритные и присоединительные размеры серии RA605

Технические характеристики серии RA605

Параметр		Значение
Число степеней свободы		6
Номинальная грузоподъемность		5 кг
Диапазон движения робота	Максимальный радиус охвата	710 мм (Точка P)
	J1	$\pm 165^\circ$
	J2	$+85^\circ \sim -125^\circ$
	J3	$+185^\circ \sim -55^\circ$
	J4	$\pm 190^\circ$
	J5	$\pm 115^\circ$
	J6	$\pm 360^\circ$
Повторяемость позиционирования		$\pm 0,02$ мм
Вес робота		40 кг
Контроллер		RCA605
Тип монтажа		Напольный, потолочный, настенный

Пульт ручного управления HIWIN

Пульт ручного управления (ПРУ) предназначен для редактирования программ и непосредственно самого программирования, управления и запоминания точек. Кроме того, в целях обеспечения безопасности пользователя пульт оснащен кнопкой аварийного выключения и включения, сделанной в виде стандартного аварийного выключателя (красная кнопка).

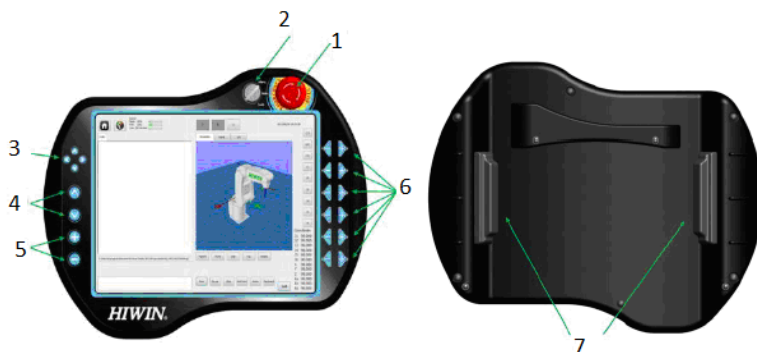


Рис. 1.3. Пульт ручного управления:
 1 – аварийная кнопка; 2 – переключатель; 3 – клавиши выбора;
 4 – эксплуатационная кнопка; 5 – изменение скорости; 6 – кнопки управления степенями свободы; 7 – переключатели разрешения

Описание, назначение и функционал частей пульта управления

Лицевая сторона:

1. Аварийная кнопка «Стоп» используется для остановки робота в чрезвычайной ситуации: он блокируется, когда кнопка аварийной остановки нажата.

!После аварийной остановки надо отжать кнопку, вращая ее по часовой стрелке.

2. Переключатель используется для изменения режима работы и включения/выключения монитора. Срабатывает только после того, как ключ вставлен в переключатель.

3. Клавиши выбора (взамен сенсорной панели).

4. Эксплуатационная кнопка.

5. Изменение скорости.

6. Кнопки управления степенями свободы в направлении часовой стрелки либо против часовой стрелки.

Тыльная (обратная) сторона:

7. Переключатели разрешения.

Примечание

Принцип работы переключателей разрешения (на обратной стороне пульта) основан на обеспечении безопасности оператора. Для управления в ручном режиме необходимо нажать до первого щелчка

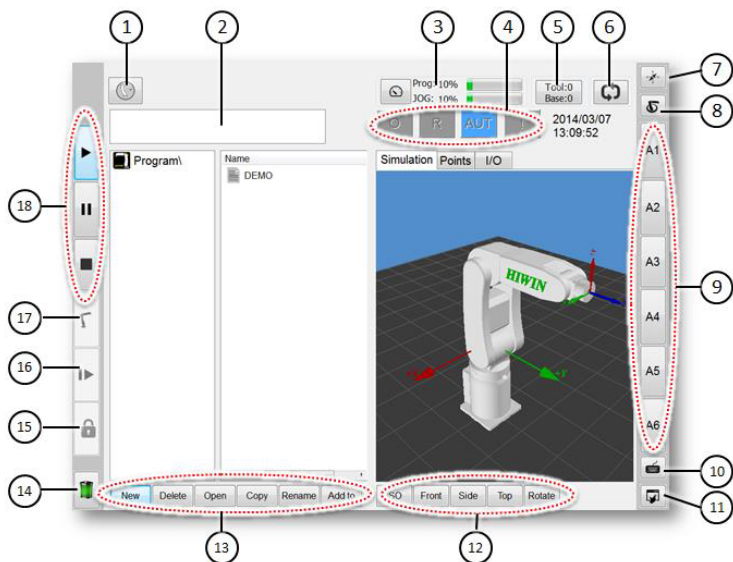
ка две клавиши, после этого станут доступны кнопки управления степенями свободы. При более сильном нажатии, до второго щелчка, робот будет остановлен, программой будет распознано опасное состояние робота для оператора.

Пульт имеет два типа управления: ручной и автоматический.

Ручной тип предназначен для управления роботом в режиме обучения, либо демонстрации, он не подойдет для управления тех-процессами по причине недостаточной точности, скорости и техно-логичности.

Автоматический же режим позволяет выполнять программы на достаточно высокой скорости с точностью повторяемости в 20 микрон. Программа в автоматическом режиме может быть загружена с помощью FLASH носителя на компьютер Hiwin и загружена с помощью программы управления.

Интерфейс пульта управления



1. Основное меню.

2. Окно, информирующее об ошибках.

3. Процент от максимальной скорости выполнения программы и управления с помощью пульта.

4. Статусная панель.
5. Выбор базы и инструмента.
6. Включение пошагового выполнения программы.
7. Установка местоположения оператора.
8. Выбор системы координат для управления с помощью кнопок.
9. Кнопки управления.
10. Кнопка отображения клавиатуры.
11. Кнопка перемещения – используется перед перемещением оператора.
12. Кнопки режимов просмотра моделирования.
13. Кнопки работы с файлами.
14. Статус аккумулятора.
15. Кнопка блокировки панели управления работой программы.
16. Следующий шаг движения.
17. Кнопка вывода манипулятора в начальное положение.
18. Кнопки управления работой программы.

Описание режимов работы робота

Режим	Применение	Скорость
T1	Используется для тестирования, программирования и обучения	Проверка программы: Запрограммированная скорость – не более 250 мм/с Ручное изменение скорости – не более чем на 250 мм/с
T2	Используется для тестирования	Проверка программы: Запрограммированная скорость. Ручное изменение скорости невозможно
AUT	Используется для работа без высокоуровневого контроллера	Режим программирования: Запрограммированная скорость. Ручное изменение скорости невозможно
AUT EXT	Используется для работа с высокоуровневым контроллером (Например ПЛК)	Режим программирования: Запрограммированная скорость. Ручное изменение скорости невозможно

Панель состояния робота

На панели отображается состояние робота (активная кнопка будет выделена цветом).



Рис. 1.4. Панель состояния робота



Подать ход декодера (кнопка горит).



Отправить отказ декодера или остановить (кнопка гаснет).

Запуск пульта управления

Главный выключатель на системе управления переключается в положение ON.

Главное меню

Нажмите **Главное меню** на пульте управления. Меню отобразится в левом поле. Щелкните элемент, чтобы отобразить таблицу функций следующего уровня (например, файл).

Функции верхнего уровня выбираются без закрытия нижних, простым нажатием на нужный элемент.

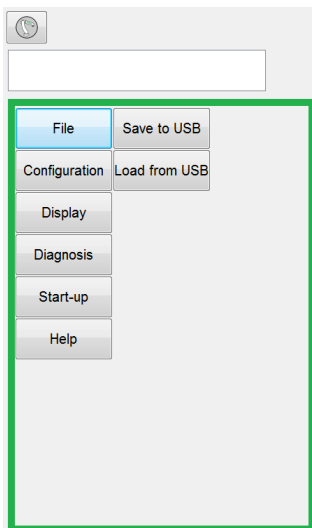


Рис. 1.5. Главное меню

Настройки групп пользователей

1. Выберите **Configuration**, затем **User group**, чтобы отобразить текущую группу пользователей

2. Нажмите **Change** для изменения текущей группы, для изменения на более высокий уровень доступа потребуется пароль.

Примечание:

Для управления в ручном режиме при включении необходимо из режима оператор переключить управление в режим ЭКСПЕРТ.

No.	Function	Operator	Engineer	Expert
	Function table			
1	File	X	X	0
2	Configuration>User group	0	0	0
3	Display	X	0	0
4	Diagnosis >Logbook	0	0	0
5	Start-up -'Calibrate	X	X	0
6	Start-up>Master	X	X	0
y	Start-up >Robot data	X	0	0
8	Start-up >Xetwork Config	X	X	0
9	Start-up>RS-232	X	X	0
	Interface			
11	Message box	0	0	0
12	Velocity configuration	X	0	0
13	Tool base coordinate	X	X	0
14	Teach Pendant configuration	X	0	0
15	Change coordinate system	X	0	0
16	JOG	X	0	0
17	Small keyboard	0	0	0
IS	Remove Teach Pendant	X	0	0
20	Program execution	0	0	0
21	Program selection	0	0	0
22	Modify program	X	X	0
23	Tool base calibration	X	0	0
24	IO operation	X	0	0
25	Functional IO modification	X	X	0

Рис. 1.6. Возможности каждой из групп пользователей

Системы координат

В системе управления робота имеются следующие системы координат: ROBOT, BASE, TOOL.

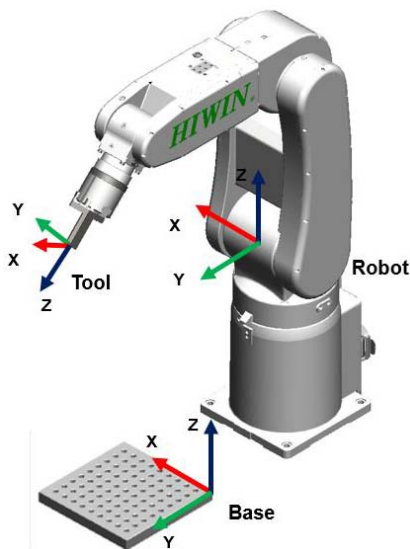


Рис. 1.7. Системы координат робота

ROBOT – система координат робота, расположенная на пересечении первого и второго суставов. Эта база по умолчанию согласуется с BASE системой координат.

BASE – базовая (неподвижная) система координат, используемая для описания положения детали.

Основанная на Robot системе координат и по умолчанию согласуется с ней.

TOOL – инструментальная система координат, находящаяся в центральной точке инструмента. По умолчанию, нулевая точка инструментальной системы координат находится в центре фланца. Смещение центра координат от фланца к инструменту задается пользователем.

Задание системы координат базовой и инструментальной

- нажать на кнопку Tool/Base;
- выбрать номер относительной системы координат инструмента или базы;

- выбрать изменяемый параметр системы координат;
- ввести имя системы координат;
- выставить значение изменяемого параметра;
- сохранить значение, нажав кнопку ОК.

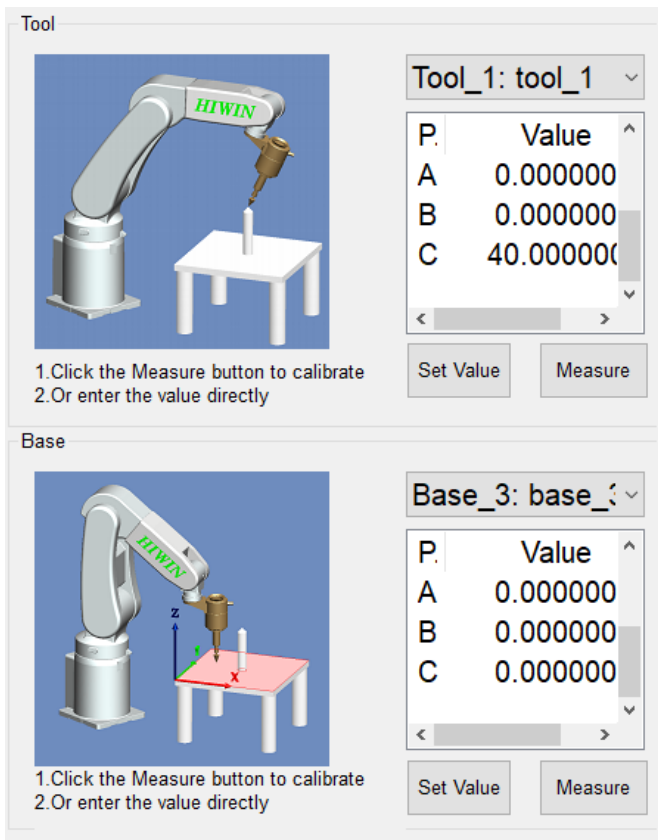


Рис. 1.8. Меню изменения системы координат

Примечание: Для использования определенной системы координат в команде необходимо после самой команды добавить «TOOL[I] BASE[J]», где I – номер системы координат инструмента, J – номер системы координат базы.

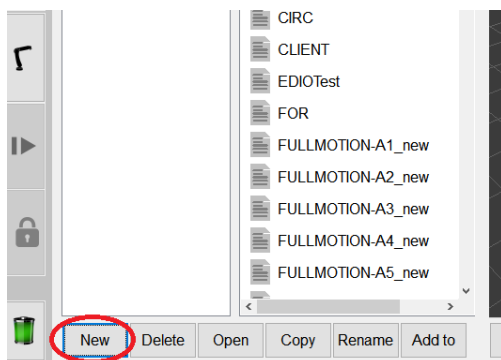
Методика выполнения работы

Порядок включения:

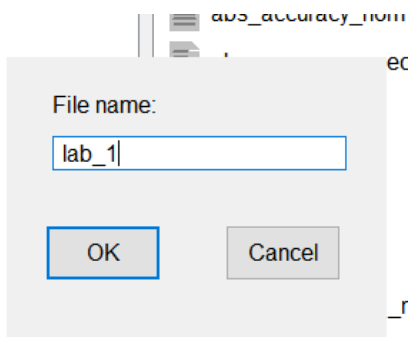
1. Включить УЗО.
 2. На задней панели блока управления роботом переключить зеленую кнопку и дождаться загрузки.
 3. Установить ключ в положение **Manu** и выбрать на пульте **T1**.
 4. Установить **режим эксперта**: настройки → **configuration** → **user group** → **expert**.
 5. Ввести пароль: **HIWIN**.
- После этого роботом можно управлять в ручном режиме.

Создание программы в режиме обучения

Нажатием кнопки **New** создаем новую программу.



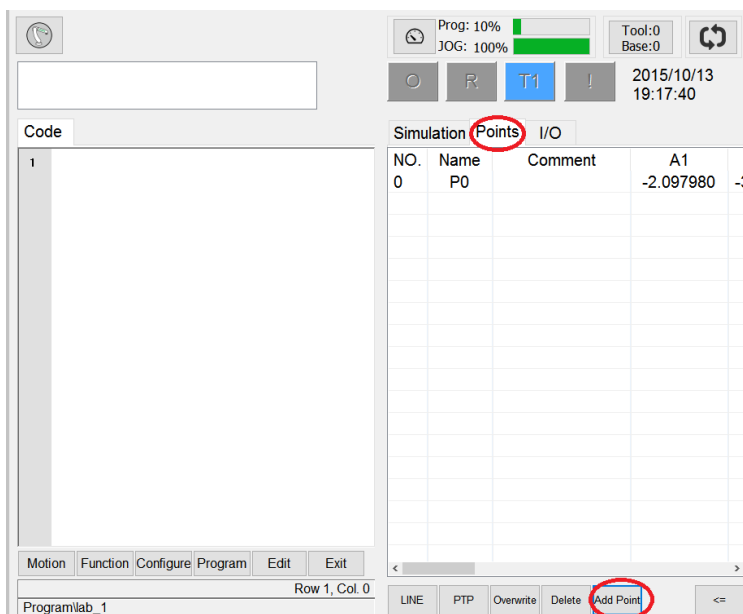
В появившемся окне вводим название программы (цифры, буквы латинского алфавита, знаки подчеркивания).



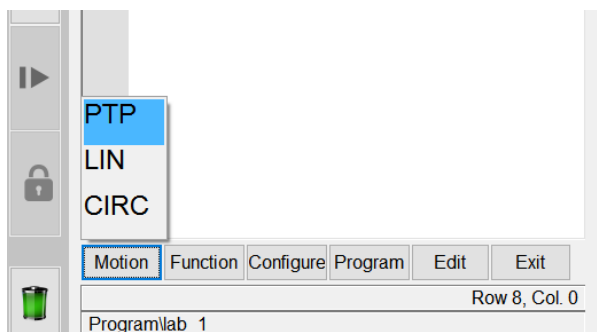
С помощью ПРУ выводим робота в первую точку, далее переходим во вкладку **Points** и нажимаем **Add Point**.

Во вкладке **Points** появилась сохраненная точка **P0**, далее с помощью ПРУ выводим робота в другие точки и также их сохраняем.

После сохранения всех точек в окне **Code** приступаем к написанию программы.



Внизу окна выбираем вкладку **Motion** и с помощью команд **PTP** или **LIN** организовываем движение между точками.



Для запуска программы зажимаем кнопку с обратной стороны пульта справа и нажимаем кнопку **Play** (▶).

С помощью вкладок **Function** и **Configure** можно организовывать более сложные программы, используя циклы, задержки и т. д.

Перенос программ во внутреннюю память робота с флеш-накопителей

1. Файлы программ требуется сохранить на флеш-накопитель в папку \HIWIN\Backup.
2. В главном меню робота выбрать File → Program → Load.
3. Выделить нужные файлы.
4. Нажать import; exit.

Задание

В режиме обучения запрограммировать робота для движения по определенной траектории в выбранной плоскости (траектория выбирается в соответствии с заданием, либо выдается преподавателем).

Варианты заданий

№	Траектория	№	Траектория
1	А	11	М
2	Б	12	Н
3	В	13	О
4	Д	14	П
5	Е	15	Р
6	Ж	16	С
7	З	17	Т
8	И	18	У
9	К	19	Х
10	Л	20	Ч

Контрольные вопросы

1. Какие элементы содержит интерфейс пульта управления?
2. Какие режимы работы имеет HIWIN RA605 и какое у них применение?
3. Какие существуют в системе управления робота системы координат?
4. Какие системы координат даны для управления с помощью кнопок пульта ручного управления и где они выбираются?
5. Как осуществляется написание программы в режиме обучения?

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Задание по варианту – траектория движения ПР с указанием контрольных точек (формат А4).
4. Листинг программы.
5. Координаты точек, сохраненных в режиме обучения.

Лабораторная работа № 2

ПРОГРАММИРОВАНИЕ ТРАЕКТОРИИ ДВИЖЕНИЯ МАНИПУЛЯТОРА

Цель работы

1. Изучить основные команды системы программирования HIWIN ROBOT SYSTEM SOFTWARE (HRSS) при создании траектории движения манипулятора.
2. Изучить особые точки ПР HIWIN.
3. Изучить особенности программирования со встроенными формами.
4. Изучить способы создания и декларирования точек.

Основные сведения

Программирование перемещений рабочей точки инструмента

Для ПР HIWIN существует возможность запрограммировать перемещение рабочей точки инструмента (РТИ) нескольких типов:

- от точки к точке (PTP);
- линейные перемещения (LIN);
- дуговое перемещение (CIRC).

Линейное и дуговое перемещение также называют непрерывным перемещением (CP motion)

Перемещение от точки к точке (PTP)

Робот перемещает инструмент к указанной точке по наиболее быстрому пути. Обычно этот путь не самый короткий, и не является прямой. Так как звенья выполняют вращательное движение, путь по кривой быстрее, чем по прямой. Траектория движения РТИ работа заранее не известна.

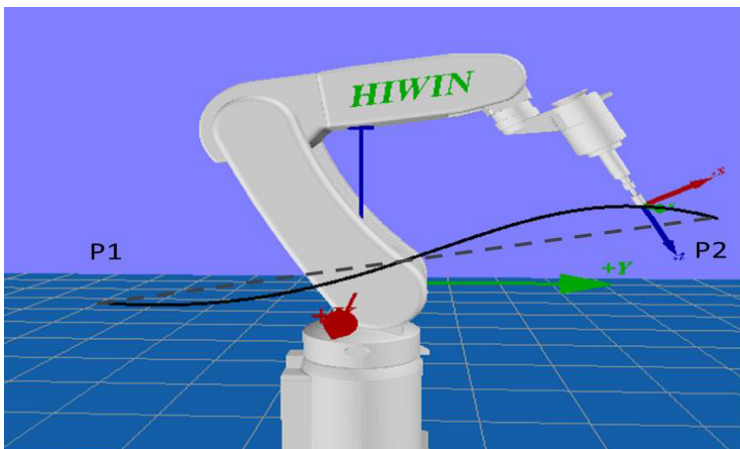


Рис. 2.1. Перемещение РТИ из точки P1 в точку P по типу RTP

Линейное перемещение (LIN)

Робот перемещает инструмент к целевой точке вдоль прямой линии с определенной скоростью.

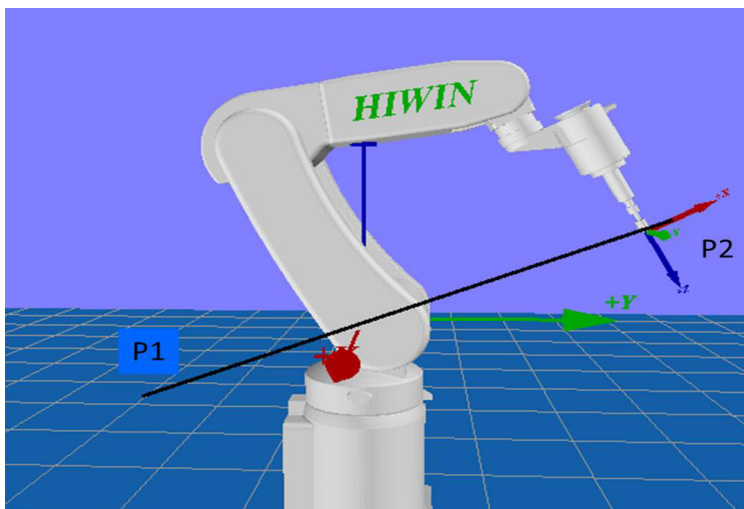


Рис. 2.2. Линейное перемещение РТИ

Контурное перемещение (CIRC)

Робот перемещает инструмент по контуру с заданной скоростью. Круговая траектория определяется начальной (текущее положение РТИ), промежуточной и конечной точками.

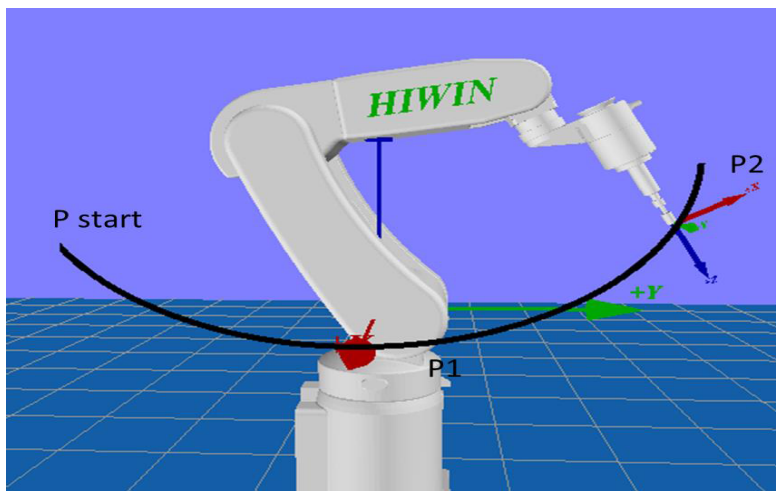


Рис. 2.3. Контурное перемещение

Сглаживание траектории (Blend)

Действие **Blend** означает, что траектория инструмента между начальной и конечной точками будет изменена, для выполнения плавного движения. Инструмент не будет перемещаться по запрограммированным точкам, которые находятся между начальной и конечной точкой. Сглаживание рассчитывается при программировании перемещения РТИ.

При перемещении от точки к точке РТР:

- инструмент изменит путь;
- конечная точка будет достигнута по максимально быстрой траектории;
- если задавать слишком много точек, то ни точка, через которую пройдет путь, ни изменение траектории не прогнозируемо.

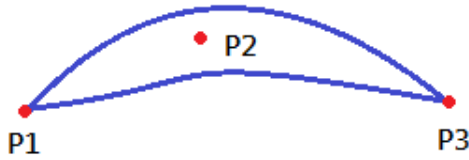


Рис. 2.4. Движение РТР из точки P1 в точку P3 и смещение точки P2

При линейном перемещении (LIN):

- инструмент изменит путь;
- конечная точка будет достигнута по кратчайшему пути;
- новая траектория не будет дугой.

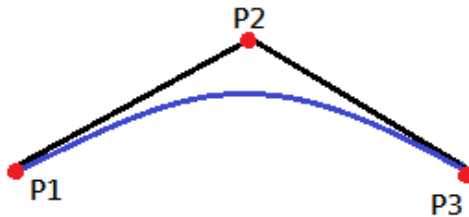


Рис. 2.5. Линейное перемещение и смещение точки P2

При контурном перемещении (CIRC):

- инструмент изменит путь;
- конечная точка будет достигнута по кратчайшему пути;
- вспомогательная точка будет пройдена точно.

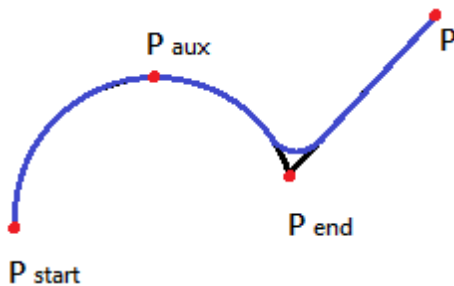


Рис. 2.6. Контурное перемещение и точка P end смещена

В коде сглаживание траектории задается параметром (CONT/FINE) после команды перемещения.

CONT – обработка траектории со сглаживанием:

PTP P0 CONT Vel=100% Acc=50% TOOL[0] BASE[0]

FINE – обработка траектории без сглаживания:

PTP P0 FINE Vel=100% Acc=50% TOOL[0] BASE[0]

Линейное и контурное перемещение

Инструмент может быть ориентирован отлично от его положения в конечной точке. Ориентация инструмента будет постепенно меняться в движении. Когда инструмент изначально ориентирован правильно, ориентация останется такой же в движении.

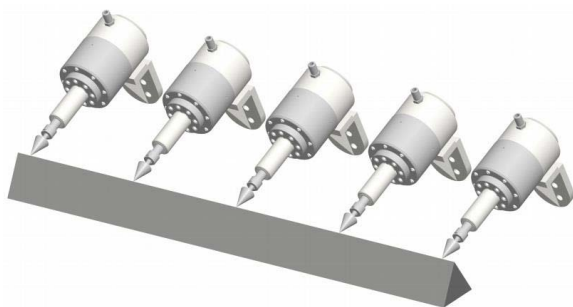


Рис. 2.7. Начальное положение в ориентации целевой точки

При контурном перемещении изменение ориентации инструмента происходит как и в случае линейного перемещения. Ориентация промежуточных точек игнорируется.

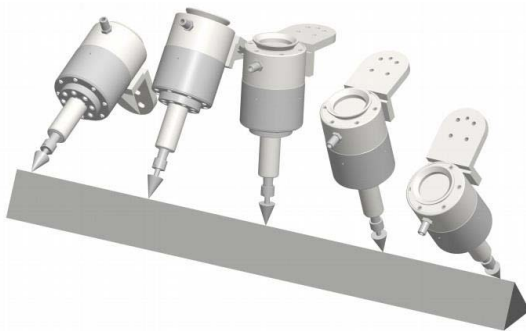


Рис. 2.8. Изменение ориентации конечной (целевой) точки относительно начальной точки траектории движения

Особые точки ПР HIWIN

ПР HIWIN с шестью степенями свободы имеет 3 вида особых точек:

- верхняя особая точка;
- особая точка в выдвинутом положении;
- особая точка на оси запястья.

Особая точка – это положение, когда минимальное изменение в декартовой системе координат может привести к большому изменению угла оси.

Верхняя особая точка

Для верхней особой точки точка запястья (средняя точка оси A5) является вертикалью к оси A1.

Положение оси A1 не может быть однозначно определено при помощи обратного преобразования, и оно может принимать любое значение.

В это время, если выполняется обратное движение, появляется ошибка.

Особая точка в выдвинутом положении

Для особой точки в выдвинутом положении точка запястья (средняя точка оси A5) находится в расширении оси A2 и A3.

Робот находится на краю рабочей области.

Хотя только один угол оси может быть получен путем обратного преобразования, малое изменение декартовой координаты вызовет большое изменение скорости оси A2 и A3.

В это время, если выполняется обратное движение, появляется ошибка.

Особая точка на оси запястья

Для особой точки оси запястья, ось A4 параллельна A6, а ось A5 находится в пределах $\pm 0,1^\circ$.

Позиции двух осей не могут быть обеспечены с помощью обратного преобразования. Хотя оси A4 и A6 могут иметь множество позиций, сумма углов оси будет та же самая.

В это время, если выполняется обратное движение, появится ошибка.

Программирование нелинейных встроенных форм

Часто используемые команды, представленные в HRSS, имеют встроенные формы. Эти команды могут упростить конструирование программы.

Примечание

Команды могут быть просто написаны в поле кода и без встроенных форм HRL (языка робота HIWIN).

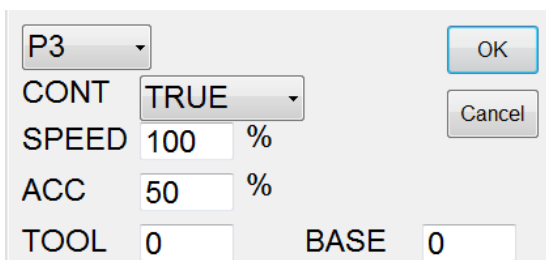
Имя в форме нелинейного движения

Программа использует строки с наименованием команд. Имя и команда должны удовлетворять следующим правилам:

- не более 30 символов;
- Специальные символы не могут быть использованы за исключением '\$-';
- строка не может начинаться с числа.

Программирование движения от точки к точке PTP Motion

- выберите Motion > PTP;
- измените соответствующие параметры (рис. 2.9);
- нажмите клавишу ОК.



P3						OK
CONT	TRUE					Cancel
SPEED	100	%				
ACC	50	%				
TOOL	0		BASE	0		

Рис. 2.9. Интерфейс встроенной формы команды PTP

Пример кода:

```
PTP P1 CONT Vel=100% Acc=50% TOOL[0] BASE[0]
```

```
PTP P1 CONT Vel=100% Acc=50% TOOL[0] BASE[0]
```

Программирование линейного движения LIN Motion

- выберите Motion > LIN;
- измените соответствующие параметры;
- нажмите клавишу ОК.

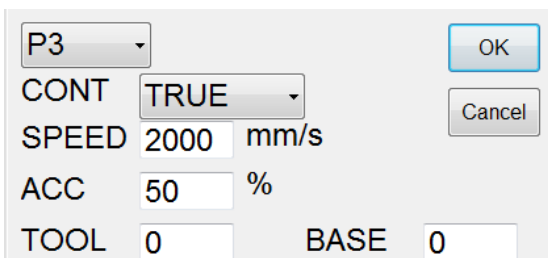


Рис. 2.10. Интерфейс встроенной формы команды LIN

Пример кода:

```
LIN P1 CONT Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Программирование кругового движения CIRC Motion

- выберите Motion > CIRC;
- в “Auxiliary” выберите вспомогательную точку;
- нажмите [SET];
- в “Actual end” выберите конечную точку;
- нажмите кнопку [SET];
- в появившемся окне введите CONT, SPEED, ACC, TOOL, BASE;
- нажмите клавишу ОК.

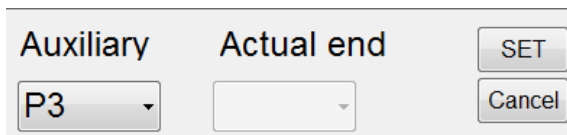


Рис. 2.11. Интерфейс встроенной формы команды CIRC

Пример кода:

```
CIRC P1 P2 CONT Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

Программное задание координат положения звеньев робота.

Смещение на заданную координату

Команда E6POS Point Coordinate

С помощью данной команды создается точка с указанными координатами.

Пример кода:

```
DECL E6POS P1 = {X 0,Y 300,Z 200} // создание точки P1  
PTR P1 CONT Vel=100% Acc=50% TOOL[0] BASE[0] // перемещение в P1
```

Направления осей базовой (декартовой) системы координат показаны на рис. 2.12.

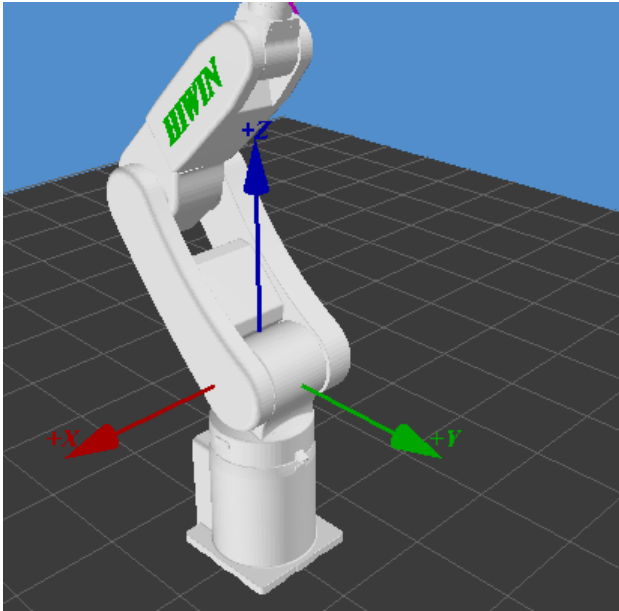


Рис. 2.12. Направления осей XYZ

Команда E6AXIS Point

С помощью данной команды создается точка, но вместо декартовых координат задаются углы поворота звеньев робота.

Пример кода:

```
DECL E6AXIS P1 = {A5 90} // создание точки P1  
PTR P1 CONT Vel=100% Acc=50% TOOL[0] BASE[0] // перемещение в P1
```

Положение робота при нулевых углах поворота звеньев (точка HOME) показано на рис. 2.13.

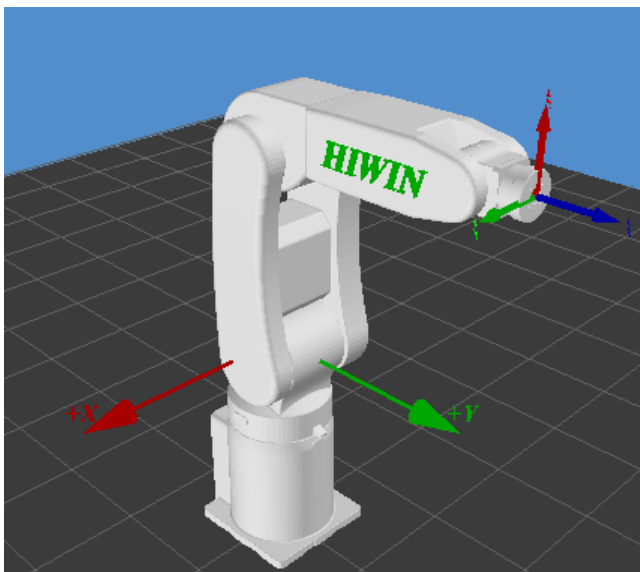


Рис. 2.13. Положение робота при нулевых углах поворота звеньев

Команда LIN_REL

С помощью данной команды задаются смещения относительно текущей точки по координатам XYZ в мм.

Пример кода:

```
LIN_REL {X 100} CONT Vel=100% Acc=50% TOOL[0] BASE[0] // смещение по оси X на 100 мм
```

Способы задания координат точек при позиционной (PTP) интерполяции

absolute – абсолютный. Привод A1 поворачивается на +45 и –45 градусов относительно абсолютной системы координат:

```
PTP {A1 45}
```

```
PTP {A1 -45}
```

relative – относительный. Привод A1 поворачивается +90 и –90 градусов относительно своего текущего положения:

```
PTP_REL {A1 90}
```

```
PTP_REL {A1 -90}
```

coordin – координатный. Перемещение в заданные координаты:

```
PTP {X -260.215295, Y 260.215295, Z 293.500000}
```

```
PTP {X 260.215295, Y 260.215295, Z 293.500000}
```

Способы задания координат точек при линейной (LINE) интерполяции

absolute – абсолютный. Перемещение РТИ на +150 мм и –150 мм относительно абсолютной системы координат:

LIN {X 150}

LIN {X –150}

relative – относительный. Перемещение РТИ на +150 мм и –150 мм относительно своего текущего положения:

LIN_REL {X 300}

LIN_REL {X –300}

Axis – осевое. Точки перемещения заданы углами поворота приводов в градусах:

LIN {A1 –22.176236, A2 –4.986184, A3 5.204342, A4 –0, A5 –90.218158}

LIN {A1 22.176236, A2 –4.986184, A3 5.204342, A4 –0, A5 –90.218158}

Способы задания координат точек при дуговой (CIRC) интерполяции

Для перемещения по дуге задаются две точки. Первая – вспомогательная точка, вторая – конечная точка. Начальная точка – текущее положение.

absolute – абсолютный. Перемещение РТИ по кругу радиусом 150 мм.

CIRC {X 0, Y 450} {X –150, Y 300}

CIRC {X 0, Y 150} {X 150, Y 300}

relative – относительный. Перемещение РТИ по кругу радиусом 150 мм.

CIRC_REL {X –150, Y 150} {X –150, Y –150}

CIRC_REL {X 150, Y –150} {X 150, Y 150}

Axis – осевое. Точки перемещения заданы углами поворота приводов в градусах.

CIRC {A1 0, A2 –14, A3 15, A4 –0, A5 –91} {A1 26, A2 5, A3 –5, A4 –0, A5 –90}

CIRC {A1 0, A2 36, A3 –25, A4 –0, A5 –101} {A1 –26, A2 5, A3 –5, A4 –0, A5 –90}

Рассмотрим пример программы движения по траектории «круг» радиусом 150 мм. Круговая траектория определена 4 точками. Точки P0 и P2 вспомогательные, а P1 и P3 конечные:

DECL E6POS P1 = {X 350, Y 368, Z 293.5}

DECL E6POS P0 = {X 200, Y 536, Z 293.5}

```
DECL E6POS P2 = {X 200, Y 218, Z 293.5}  
DECL E6POS P3 = {X 50, Y 386, Z 293.5}  
PTP P3  
CIRC P0 P1 CONT Vel=2000mm/s Acc=50% TOOL[0] BASE[0]  
CIRC P2 P3 CONT Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
```

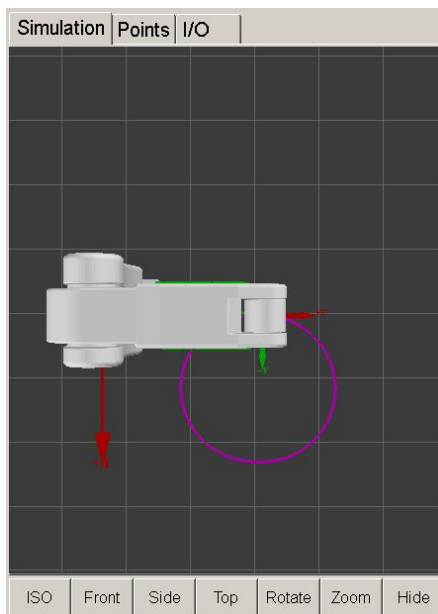


Рис. 2.14. Результат выполнения программы «отрисовки» круга РТИ

Задание

Написать и запустить на выполнение в режиме T1 программу движения ПР по траектории согласно варианту из лабораторной работы № 1.

Начальную и конечную точки задекларировать двумя разными способами. Перемещение должно выполняться следующим образом:

- из точки HOME в режиме PTP на оптимальной скорости;
- по точкам траектории – в приращениях по осям XYZ на рабочей скорости в режимах LINE или CIRC в зависимости от типа линий различных участков траектории.

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Задание.
4. Листинг программы.

Лабораторная работа № 3

АНАЛИТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ ПРОМЫШЛЕННОГО РОБОТА HIWIN

Цель работы

1. Изучить переменные и операторы, а также вызов и использование их в HRSS.
2. Изучить встроенные формы основных команд языка программирования промышленного робота HIWIN.
3. Получить практические навыки аналитического программирования промышленного робота HIWIN.

Основные сведения об используемых переменных системы программирования HRSS

Переменная вещественного типа

REAL

Пример:

REAL One

One = 1

Формат подобен типу десятичных данных. После закрытия программы значение переменной исчезнет.

Возможность сохранения 32 бит. Можно сохранить только 6 первых десятичных разрядов.

Переменная целого типа

INT

Пример:

INT Two = 2

Формат подобен целочисленному типу данных. После закрытия программы значение переменной исчезнет.

Возможность сохранения 32 бит, диапазон хранения от 2147483648 до 2147483647.

Переменная логического типа

BOOL

Пример:

BOOL K = TRUE

Логическая переменная. После закрытия программы значение переменной исчезнет.

Переменная должна быть объявлена как истина (TRUE) или ложь (FALSE).

Переменная символьного типа

CHAR

Пример:

CHAR COLOR = 'R'

Переменные байты, после закрытия программы, значение переменной исчезнет.

Переменная должна быть объявлена как указанный байт.

Объявление переменных с помощью встроенных форм

Рассмотрим на примере переменных вещественного типа:

Выберите пункт Configure > Variable > REAL.

Установите параметры в интерфейсе (рис. 3.1).

Сохраните команду, нажав кнопку ОК.

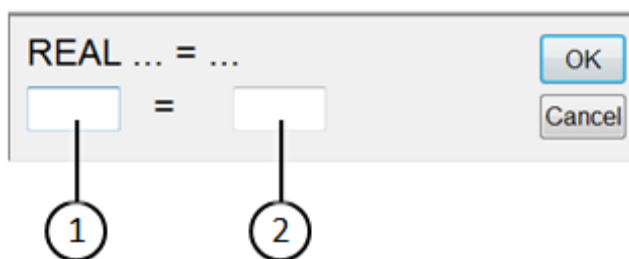


Рис. 3.1. Интерфейс встроенной формы переменной вещественного типа REAL:
1 – имя переменной; 2 – начальное значение

Организация временной задержки движения ПР

Команда WAIT

Для остановки манипулятора на определенное время используется задержка по принципу ЖДАТЬ – ПОКА, выраженная в секундах.

Выберите FUNCTION > WAIT FOR.

Установите параметры в интерфейсе (рис. 3.2)

Сохраните команду, нажав кнопку ОК.

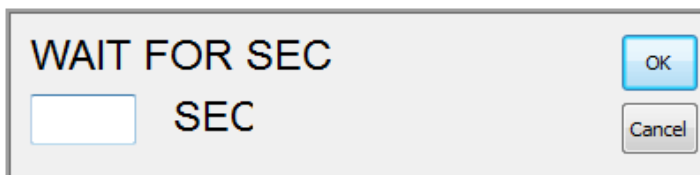


Рис. 3.2. Интерфейс встроенной формы команды WAIT FOR SEC

Операции условного и безусловного перехода

Команда GOTO

IF *условие* THEN

GOTO LABEL1

ENDIF

IF *условие* THEN

GOTO LABEL 2

ENDIF

IF *условие* THEN

GOTO LABEL 3

ENDIF

LABEL 1:

.....

LABEL 2:

.....

LABEL 3:

.....

Команда ЕСЛИ (IF)

Формат 1

IF *условие* THEN

.....

ENDIF

Формат 2

IF *условие* THEN

.....

ELSE

.....

ENDIF

С использованием выражений истинности и логических выражений (Вариант 1).

IF ((TRUE) AND (TRUE)) THEN

.....

ENDIF

С использованием выражений истинности и логических выражений (Вариант 2).

IF ((TRUE) OR (FALSE)) THEN

.....

ENDIF

Для упрощения процесса создания аналитических программ, используются встроенные формы:

Выберите вкладку FUNCTION > IF ENDIF > значение.

Установите параметры в интерфейсе (рис. 3.3).

Сохраните команду, нажав кнопку ОК.



Рис. 3.3. Интерфейс встроенной формы команды IF

Множественный выбор – селектор

Функция SWITCH

SWITCH без варианта невыполнения условия:

SWITCH *число*

.....

CASE *число1*

.....

CASE *число2*

.....

ENDSWITCH

SWITCH с вариантом невыполнения условия:

SWITCH *число*

.....

CASE *число1*

.....

CASE *число2*

.....

DEFAULT

EXIT

ENDSWITCH

С использованием переменных символьного типа

SWITCH *символ*

.....

CASE *символ1*

.....

CASE *символ2*

.....

DEFAULT

EXIT

ENDSWITCH

Циклические алгоритмы

Циклы с параметром

Команда FOR

Формат команды:

FOR начальное объявление переменной TO последнее значение STEP
значение дискреты

.....
ENDFOR

Циклы с условием

Команда WHILE

Формат команды:

WHILE *условие*

.....
ENDWHILE

С использованием выражений истинности и логических выражений:

WHILE ((TRUE) AND (TRUE))

.....
ENDWHILE

Функция REPEAT

Формат команды:

REPEAT

.....
UNTIL *условие*

С использованием выражений истинности и логических выражений:

REPEAT

.....
UNTIL((FALSE) OR (TRUE))

Использование встроенных форм рассмотрим на примере команды WHILE:

- выберите FUNCTION > WHILE ENDWHILE > значение;
- установите параметры в интерфейсе (рис. 3.4);
- сохраните команду, нажав кнопку ОК.

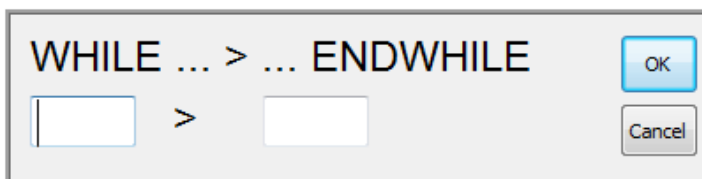


Рис. 3.4. Интерфейс встроенной формы команды WHILE

Задание

Написать программу перемещения манипулятора по траектории согласно варианту из лабораторной работы № 1. Программа должна содержать временные задержки для остановки ПР, условные переходы в разные части программы, выход в HOME по значению заданного параметра. Организовать циклическую работу ПР. Число повторов не более 5.

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Задание.
4. Листинг программы.
5. Схема алгоритма.

Лабораторная работа № 4

ИЗУЧЕНИЕ ВОЗМОЖНОСТЕЙ ТАЙМЕРОВ, СЧЕТЧИКОВ ПР HIWIN RA605

Цель работы

Изучить принцип действия таймеров, счетчиков, а также вызов и использование их в HRSS.

Основные сведения

HRSS (HIWIN Robot System Software) – программное обеспечение, отвечающее за управление ПР HIWIN RA605.

HRSS включает в себя:

- управление пользователями;
- редактор программы;
- язык программирования робота;
- встроенные формы для программирования;
- дисплей сообщения;
- окна конфигурации и т. д.

Таймер (англ. timer < time – время) – прибор производственно-технического, военного или бытового назначения, в заданный момент времени выдающий определенный сигнал, либо включающий/выключающий какое-либо оборудование через свое устройство коммутации электроцепи. Большой частью под таймерами подразумеваются устройства, отмеряющие заданный интервал времени с момента запуска. В ПР HIWIN двадцать таймеров, при помощи которых можно измерять время того или иного процесса, а также ввести временную составляющую в программу. Последнее подразумевает под собой включение таймера в тело цикла и выполнения заданных действий в течение промежутка времени. Цикл может быть не завершен в том случае, когда по истечении заданного промежутка времени не выполнено(ы) действие(я) в этом цикле. Цикл завершается после выполнения действий.

Счетчик (Counter) в данном случае не устройство подсчета, а 32-битный регистр, хранящий значение переменной программы. В ПР HIWIN таких счетчиков двадцать. Значение переменной, хранящееся

ся в регистре, не сбрасывается после закрытия программы и может использоваться другими программами.

Использование счетчика

Выберите пункт `Configure > Variable > Counter`.

Установите параметры в интерфейсе (рис. 4.1).

Сохраните команду, нажав кнопку `OK`.

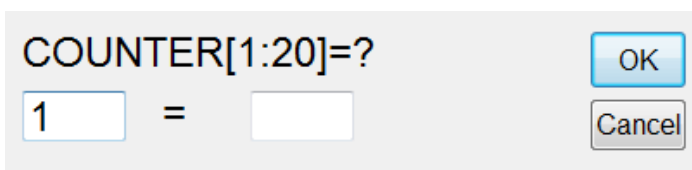


Рис. 4.1. Интерфейс встроенной формы организации счетчика COUNTER

Использование таймера

Выберите пункт `Configure > Variable > Timer`.

Установите параметры в интерфейсе (рис. 4.2).

Добавьте " `$ T_STOP [n] = False` " в строке с которой необходимо осуществить включение таймера; n – номер таймера.

Добавьте " `$ T_STOP [n] = TRUE` " в строке с которой необходимо осуществить остановку таймера; n – номер таймера.

Сохраните команду, нажав кнопку `OK`.

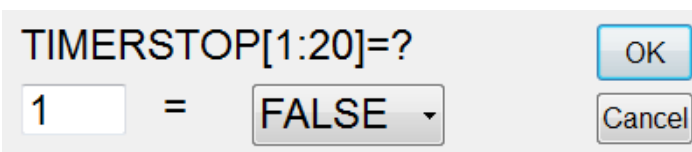


Рис. 4.2. Интерфейс встроенной формы организации таймера TIMER

Пример использования таймера в программе

Программа обеспечивает перемещение манипулятора в исходную точку, после чего ПР движется по траектории окружности с сохранением времени прохождения этой траектории.

`$T[1] = 0` установка значения таймера в 0

`$C[1] = 0` установка значения счетчика в 0

PTP {A1 0, A2 0, A3 0, A4 0, A5 -90, A6 0} CONT Vel=100%
Acc=50% TOOL[0] BASE[0]

WAIT SEC 0

\$T_STOP[1] = FALSE

CIRC {X 200, Y 536, Z 293.5} {X 350, Y 368, Z 293.5} CONT
Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

CIRC {X 200, Y 218, Z 293.5} {X 50, Y 386, Z 293.5} CONT
Vel=2000mm/s Acc=50% TOOL[0] BASE[0]

WAIT SEC 0

\$T_STOP[1] = TRUE

конец счета таймера

\$C[1] = \$T[1]

перенос значения таймера в счетчик

Примечание

Требуется обязательно добавлять перед запуском и остановкой таймера ожидание равное 0 сек, чтобы остановить предварительное чтение программы.

Для просмотра текущего значения таймера и счетчика необходимо перейти в главное меню:

НАСТРОЙКИ > Display > Variable > Counter или Timer.

Также в меню Display можно посмотреть текущее значение координат и углов приводов в пункте Actual Position.

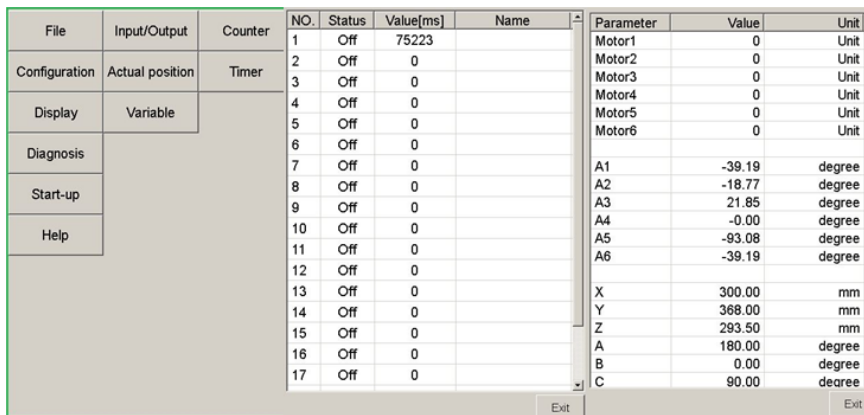


Рис. 4.3. Путь в меню, окна просмотра текущих значений таймеров и текущего положения манипулятора

Задание

Написать программу, в которой измеряется время перемещения РТИ манипулятора из исходной точки в точку **P0 {X 300, Y 368, Z 293.5}** при различных типах интерполяции.

ВАРИАНТ 1

Интерполяция позиционная (PTP). Для измерения времени использовать таймер 3. Значение времени по окончании движения записать в счетчик 15.

ВАРИАНТ 2

Интерполяция линейная (LINE). Для измерения времени использовать таймер 1. Значение времени по окончании движения записать в счетчик 11.

ВАРИАНТ 3

Интерполяция дуговая (CIRC). Для измерения времени использовать таймер 2. Значение времени по окончании движения записать в счетчик 10. Координаты точки вспомогательной точки – **P1 {X 150, Y 536, Z 293.5}**

Контрольные вопросы

1. Какие есть способы задания точек при разных видах интерполяции?
2. Каким способом задается точка без привязки к интерполяции?
3. Сколько счетчиков в ПР HIWIN?
4. Сколько таймеров в ПР HIWIN?
5. Как осуществляется организация временной составляющей в программном коде?

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Задание
4. Листинг программы.

Лабораторная работа № 5

ИЗУЧЕНИЕ ЦИФРОВЫХ ВХОДОВ-ВЫХОДОВ ПР HIWIN RA605

Цель работы

Изучить цифровые входы-выходы ПР HIWIN RA605, их вызов и использование в HRSS.

Основные сведения

Линии входов-выходов – это обычные транзисторные ключи, функции которых аналогичны функциям реле или микропереключателя: ключ включен – на внешнее устройство подано напряжение, ключ выключен – внешнее устройство отключено. Такими внешними устройствами могут быть:

- выключатели;
- датчики;
- сигнальные лампы;
- клапаны гидро- и (или) пневмосистем;
- программируемая логика;
- ЭВМ и другие роботы.

Линии гальванически развязаны, для внешних подключений требуется вспомогательное напряжение. Пользователю необходимо предусмотреть внешний источник напряжения на 24 В постоянного тока. Суммарная потребность в мощности определяется требуемой мощностью устройств, подключаемых к входным линиям, а также мощностью для управления входными линиями.

Входные сигналы должны быть типа ON/OFF, замыкающий или размыкающий контакт или подобное решение с помощью полупроводника. Оборудование данного типа включает:

- кнопки и выключатели;
- бесконтактные датчики;
- конечные переключатели;
- фотоэлементы;
- контакты реле;
- цифровые выходы программируемых логических контроллеров.

Выходные сигналы также представляют собой сигналы типа ON/OFF. С точки зрения функционирования выходы представляют собой полупроводниковые выключатели. С их помощью можно управлять работой:

- сигнальных лампочек;
- реле и разъемов;
- магнитных клапанов;
- цифровых входов программируемых логических контроллеров.

В контроллере ПП HIWIN RA605 имеется 48 цифровых входов (DI) и 48 цифровых выходов (DO). Система программирования HRSS позволяет симулировать значения на входах выходов для отладки программы или ее воспроизводства в связи с отсутствием подключенного устройства. Для этого необходимо перейти во вкладку I/O. Внизу находятся вкладки переключения между входами-выходами DI (digital input), DO (digital output), RI (remote input), RO (remote output), FI (functional input), FO (functional output). Для симуляции того или иного входа-выхода необходимо поставить галочку в квадратик столбца SIM (simulation), а для установки значения необходимо поставить галочку в квадратик столбца Value. Галочка в столбце Value означает единицу, отсутствие – ноль.

Система управления может управлять максимум 24 цифровыми входами и 24 цифровыми выходами. Конфигурация меняется в зависимости от требований пользователя.

Вход-выход может управляться с помощью следующих переменных:

	Вход	Выход
Номер	\$DI[1] ... \$DI[24]	\$DO[1] ... \$DO[24]
Номер	--	\$VO[1] ... \$VO[3]
Номер	\$RI[1] ... \$RI[8]	\$RO[1] ... \$RO[8]

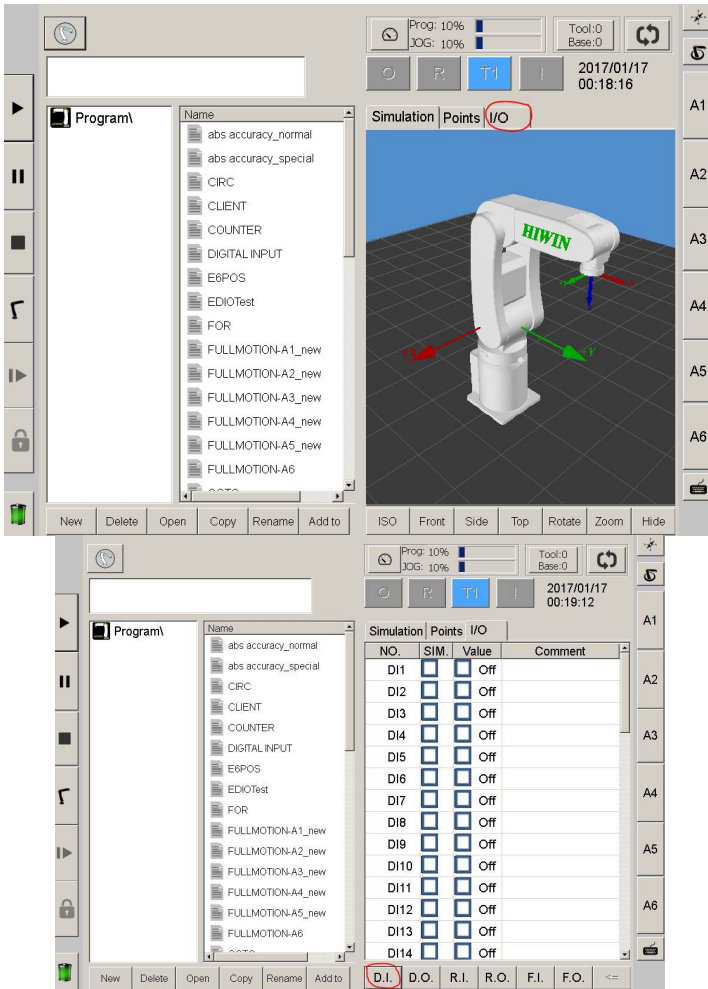


Рис. 5.1. Расположение вкладок цифровых входов и выходов

Управление по выходным цифровым линиям

Установите курсор, и вставьте следующую команду.

Выберите пункт Configure > Output > Digital.

Установите параметры в функции.

Сохраните команду, нажав кнопку ОК.

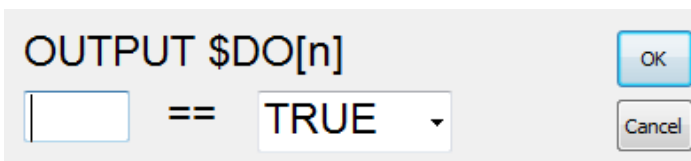


Рис. 5.2. Интерфейс встроенной формы выходов

Организация функции ЖДАТЬ ПОКА по сигналу от цифровой входной линии

Установите курсор и вставьте следующую команду.

Выберите FUNCTION > INPUT.

Установите параметры в интерфейсе (рис. 5.3).

Сохраните команду, нажав кнопку ОК.

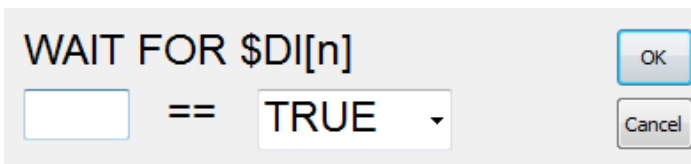


Рис. 5.3. Интерфейс встроенной формы WAIT FOR

Рассмотрим условный переход «ЕСЛИ по цифровому входу 1 «1» ТО переход на метку STEP1, ИНАЧЕ продолжение программы конец условия» конструкция IF IN:

```
IF $DI[1] == TRUE THEN
```

```
GOTO STEP1
```

```
STEP_1:
```

```
ENDIF
```

Пример использования цифровых входов-выходов

Рассмотрим программу, по которой ПР ждет единицы на входах 1–2, а затем двигается в точку P1, ждет ноль на входах 1–4 и двигается по линии к точке P2, при завершении записывается +1 к значению в счетчике 1. LOOP – бесконечный цикл.

```

INT I
$C[1] = 0                обнуляем счетчик
LOOP
I = 20
FOR I = 1 TO 2          для входов 1–2 ждем единицы
WAIT FOR $DI[I] == TRUE
ENDFOR
WAIT SEC 0.5
LIN P1 FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
FOR I = 1 TO 4         для входов 1–4 ждем нуля
WAIT FOR $DI[I] == FALSE
ENDFOR
WAIT SEC 0.5
$C[1] = $C[1] + 1      прибавляем 1 к значению в счетчике 1
LIN P2 FINE Vel=2000mm/s Acc=50% TOOL[0] BASE[0]
ENDLOOP

```

Задание

Написать и отработать в режиме T1 программу перемещения манипулятора по траектории, согласно варианту из лабораторной работы № 1. Предусмотреть переходы по условию срабатывания цифровых ключей и включение индикаторов на цифровых линиях выхода.

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Задание.
4. Листинг программы.

Контрольные вопросы

1. Приведите пример устройств, которые могут быть подключены к цифровым входам.
2. Приведите пример устройств, которые могут быть подключены к цифровым выходам.

Лабораторная работа № 6

ВВЕДЕНИЕ В РАБОТУ С ПРОМЫШЛЕННЫМ КОЛЛАБОРАТИВНЫМ РОБОТОМ-МАНИПУЛЯТОРОМ UNIVERSAL ROBOTS UR3

Цель работы

Ознакомиться с интерфейсом пульта управления роботом.

Основные сведения

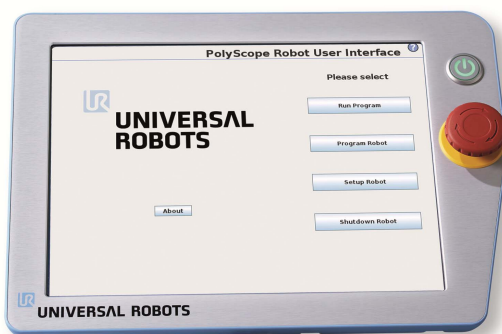


Рис. 6.1. Пульт управления (Touch sensitive monitor)

Быстрый запуск

1. Нажмите кнопку питания на подвесном пульте обучения.
2. Подождите минуту, пока система запускается; на сенсорном экране будет отображаться текст.
3. Когда система будет готова, на сенсорном экране отобразится всплывающее окно с сообщением о необходимости проведения инициализации робота.
4. Коснитесь кнопки во всплывающем окне. Будет открыт экран инициализации.
5. Подождите появления диалогового окна **«Подтверждение примененной конфигурации безопасности»** и нажмите кнопку **«Подтвердить конфигурацию безопасности»**. При этом применяется начальный набор параметров безопасности, которые должны быть скорректированы на основе оценки рисков.

6. Выйдите из зоны досягаемости (рабочей зоны) робота.
7. Нажмите кнопку **Вкл.** во всплывающем окне. Подождите несколько секунд, пока состояние робота изменится на **Бездействие**.
8. Убедитесь, что масса полезной нагрузки и выбранный монтаж корректны. В случае обнаружения расхождений между выбранным монтажом и данными датчика будет выведено предупреждение.
9. Нажмите кнопку **Запуск** во всплывающем окне. Робот издаст звук и немного сместится при отключении тормозов.
10. Нажмите кнопку **ОК**, чтобы перейти на экран приветствия.

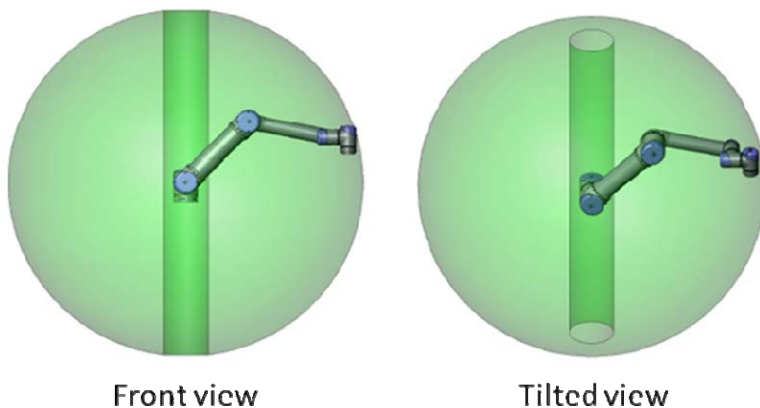


Рис. 6.2. Рабочая зона манипулятора UR3

Для создания простой программы на запущенном роботе выполните следующие действия:

1. Нажмите кнопку **Запрограммировать робота** и выберите пункт **Очистить программу**.
2. Нажмите кнопку **Далее** (справа внизу), чтобы выбрать строку в структуре дерева в левой части экрана.
3. Перейдите на вкладку **Структура**.
4. Нажмите кнопку **Переместить (Move)**.
5. Перейдите на вкладку **Команда**.
6. Нажмите кнопку **Далее**, перейдите к настройкам контрольной точки (Waypoint).
7. Нажмите кнопку **Установить эту контрольную точку** рядом со значком «?».

8. На экране **Переместить (Move)** переместите робота, нажимая различные синие стрелки, или удерживайте нажатой кнопку **Свободный привод** (дублируется сзади пульта), расположенную на задней стороне подвесного пульта обучения, вручную перемещая манипулятор робота.

9. Нажмите **ОК**.

10. Нажмите **Добавить контрольную точку до**.

11. Нажмите кнопку **Установить эту контрольную точку** рядом со значком «?».

12. На экране **Переместить** переместите робота, нажимая различные синие стрелки, или удерживайте нажатой кнопку **Свободный привод**, вручную перемещая манипулятор робота.

13. Нажмите **ОК**.

14. Программа готова. При нажатии символа **Воспроизвести** робот будет перемещаться между двумя заданными точками. Отойдите на безопасное расстояние, положите руку на кнопку аварийной остановки и нажмите **Воспроизвести**.

15. Поздравляем! Вы создали первую программу робота, по которой робот перемещается между двумя заданными контрольными точками.

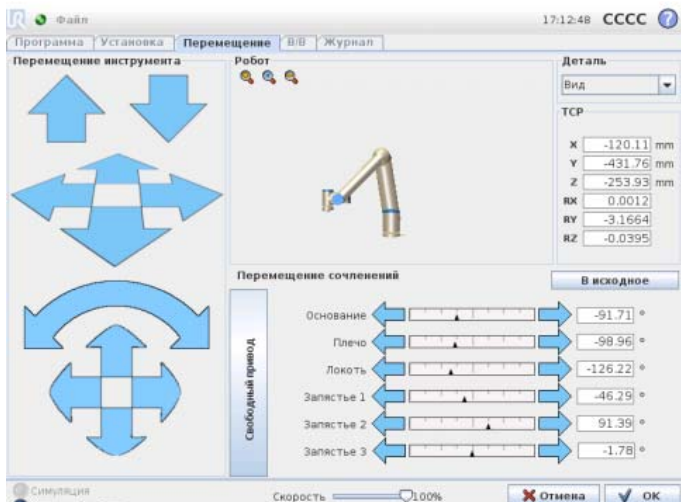


Рис. 6.3. Вкладка Переместить



Рис. 6.4. Экран приветствия

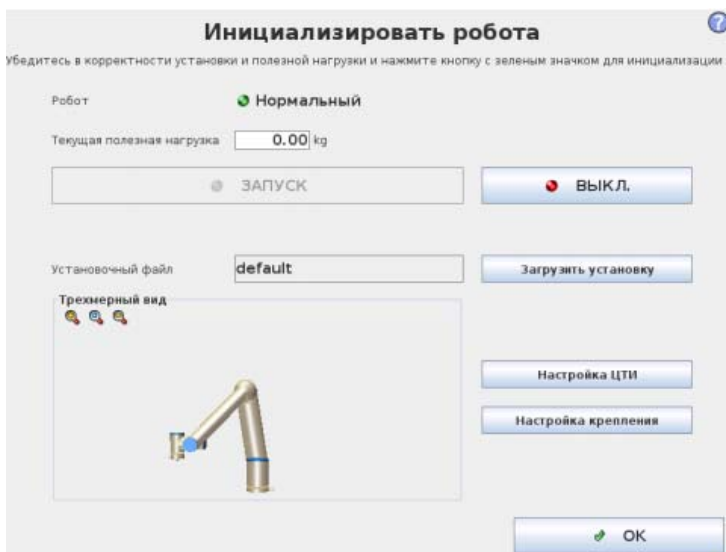


Рис. 6.5. Экран инициализации

Вкладка **Автоперемещение** используется, если робот должен перейти в указанное положение в пределах своей рабочей зоны. Например, если необходимо переместить манипулятор робота в исходное положение программы перед запуском или переместить в контрольную точку при изменении программы.

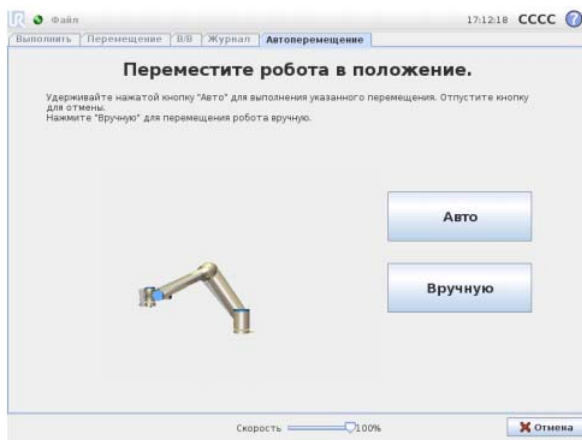


Рис. 6.6. Вкладка Автоперемещение

Типы движений

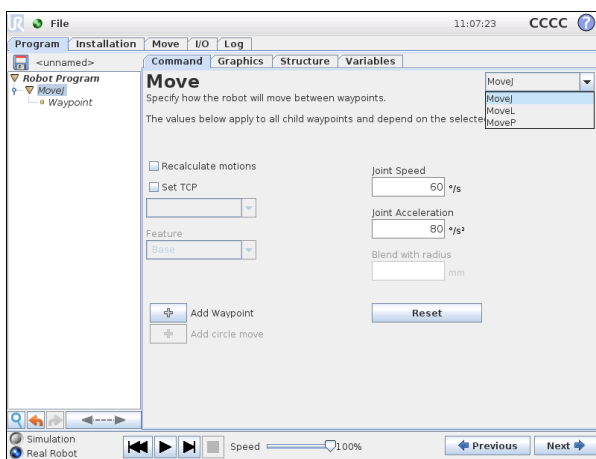


Рис. 6.7. Вкладка Command – Move

Команда Move

Команда Move используется для управления движением робота с помощью ключевых контрольных точек.

Контрольные точки должны быть включены в команду Move.

Команда Move определяет ускорение и скорость перемещения манипулятора робота между этими контрольными точками.

Можно выбрать один из трех типов движения:

- MoveJ;
- MoveL;
- MoveP.

Описание каждого типа движения приводится далее.

MoveJ используется для выполнения движений, которые рассчитаны в зоне сочленений манипулятора робота. При управлении сочленениями каждое из них одновременно достигает заданного конечного положения.

При таком типе движения траектория инструмента имеет форму кривой. Для этого типа движений используются такие общие параметры, как максимальная скорость сочленения и ускорение сочленения, которые применяются при расчетах движений, указанных в мм/с и мм/с² соответственно.

Если требуется, чтобы манипулятор робота быстро перемещался между контрольными точками независимо от траектории инструмента между этими контрольными точками, рекомендуется выбрать этот тип движений.

MoveL используется для линейного перемещения инструмента между контрольными точками. Это означает, что каждое сочленение выполняет более сложное движение, чтобы инструмент оставался на прямой траектории.

Для этого типа движений можно задать такие общие параметры, как требуемая скорость инструмента и ускорение инструмента, указанные в мм/с и мм/с² соответственно, а также деталь. Выбранная деталь будет определять, в зоне какой детали будут находиться положения инструмента для контрольных точек.

Особый интерес в плане зон деталей представляют переменные детали и переменные контрольные точки.

Переменные детали можно использовать, если положение инструмента для контрольной точки необходимо определить исходя

из фактического значения переменной детали при выполнении программы робота.

MoveP используется для линейного перемещения инструмента с постоянной скоростью при выполнении круговых движений и предназначен для выполнения ряда операций обработки, например нанесения клея или распыления.

Величина радиуса круговых движений по умолчанию является общим значением для всех контрольных точек. Чем меньше значение, тем более острым будет угол поворота траектории, тогда как при более высоком значении траектория будет более плавной.

При движении манипулятора робота по контрольным точкам с постоянной скоростью блок управления роботом не может ждать операции ввода-вывода или действия оператора. При этом движение манипулятора робота может быть прервано или произойдет защитная остановка.

Возможно добавление переменной Circle Move к команде moveP, состоящей из двух контрольных точек: первая точка задает промежуточную точку на дуге окружности, а вторая точка является конечной точкой перемещения.

Робот начнет круговое движение из текущего положения и пройдет через две заданные контрольные точки.

Изменение ориентации инструмента в ходе кругового движения определяется только начальной ориентацией и ориентацией в конечной точке, поэтому ориентация в промежуточной точке не влияет на круговое движение.

Переменной Circle Move должна всегда предшествовать контрольная точка в том же разделе MoveP.

Команда Waypoint (фиксированное положение)

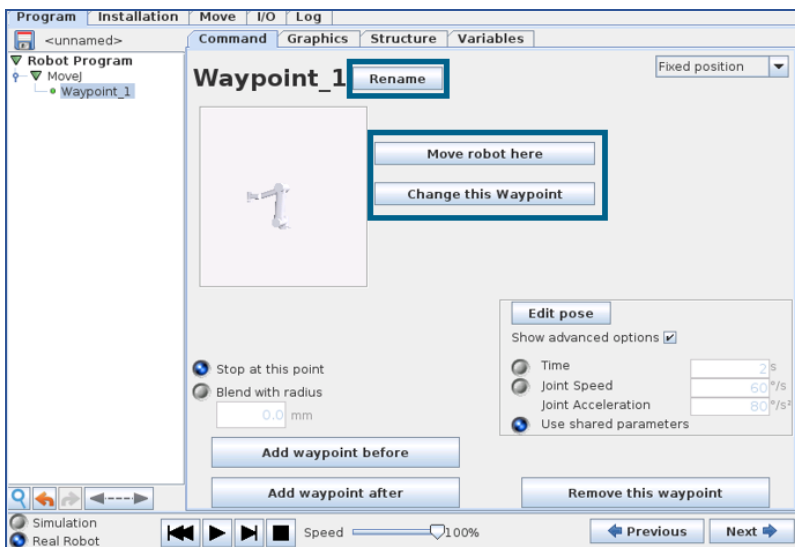


Рис. 6.8. Вкладка Команда – Контрольная_точка

Точка на траектории робота. Контрольные точки – это самая важная часть программы манипулятора робота. Они указывают роботу, где он должен находиться.

Для установки контрольной точки с фиксированным положением необходимо физически переместить манипулятор в это положение.

Установка контрольной точки

Нажмите кнопку, чтобы открыть экран Перемещение, на котором можно указать положение манипулятора робота для данной контрольной точки.

Если контрольная точка включается в команду Move в линейной зоне (MoveL или MoveP), должна быть указана допустимая деталь для этой команды Move, чтобы можно было нажать кнопку.

Радиус круговых движений

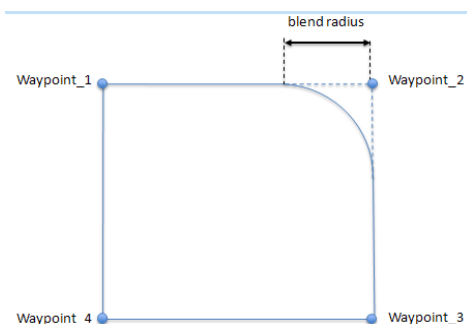


Рис. 6.9. Сглаживание траектории вблизи точки Waypoint_2

Если задан радиус скругления, то траектория движения манипулятора робота плавно обогнет контрольную точку, не давая манипулятору робота остановиться в ней.

Круговые движения не должны пересекаться, поэтому не удастся установить радиус круговых движений, который будет пересекаться с радиусом круговых движений предыдущей и следующей контрольной точки. Точкой останова является контрольная точка с радиусом круговых движений 0,01 мм.

Команда Waypoint (Относительное положение)

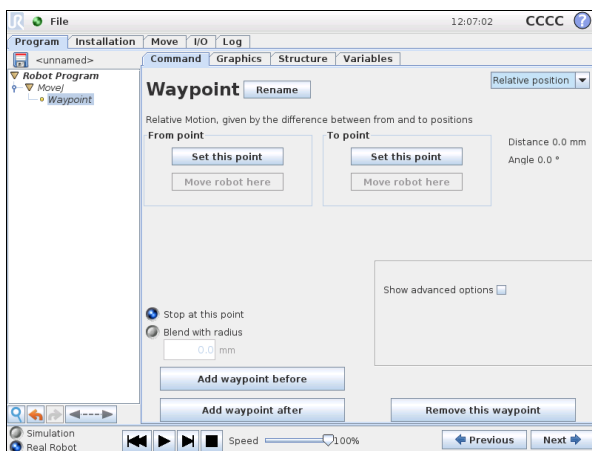


Рис. 6.10. Команда Waypoint (Относительное положение)

Контрольная точка, положение которой задано относительно предыдущего положения манипулятора робота, например, «на два сантиметра левее».

Относительное положение определяется разностью между двумя заданными положениями (слева или справа).

Обратите внимание, что при использовании нескольких относительных положений подряд манипулятор робота может выйти из своей рабочей зоны.

В данном случае под расстоянием понимается расстояние в прямоугольной системе координат между ЦТИ в двух положениях. Угол показывает, насколько изменяется ориентация ЦТИ между этими двумя положениями.

Более точное описание изменения ориентации дает длина вектора вращения.

Задание

Построить траекторию движения манипулятора в соответствии с вариантом, не используя «свободный привод».

Вариант	1	2	3	4
Фигура	Квадрат	Параллелограмм	Ромб	Круг

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Задание.
4. Листинг программы.

Лабораторная работа № 7

РАБОТА С РАБОЧИМ ОРГАНОМ МАНИПУЛЯТОРА UR3

Цель работы

Изучить возможности использования схвата.

Основные сведения

Рабочий орган

На конце манипулятора (на его «Запястье», рис. 7.1 позиция Wrist3) располагается рабочий орган – устройство, предназначенное для выполнения специального задания. В качестве рабочего органа может выступать захватное устройство или технологический инструмент.

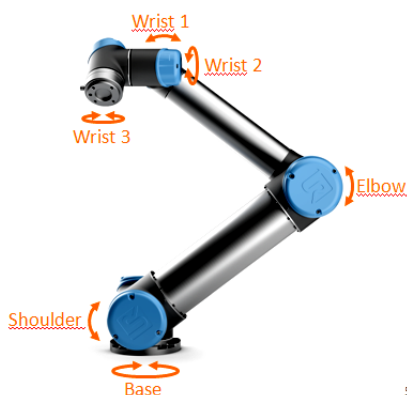


Рис. 7.1. Манипулятор UR3

Наиболее универсальной разновидностью захватного устройства является схват – устройство, в котором захватывание и удержание объекта производится посредством относительного перемещения частей данного устройства.

По способу удержания объекта захватные устройства подразделяют на:

– схватывающие (механические схваты и устройства с эластичными рабочими камерами, в которые нагнетают жидкость или сжатый воздух);

– поддерживающие (в них объект не зажимают, а применяют для его удержания нижнюю поверхность, выступающие части объекта или имеющиеся в нем отверстия);

– удерживающие (в них на объект оказывают силовое воздействие за счет различных физических эффектов: вакуумные, магнитные и электростатические захваты, адгезия и т. п.).

Число применений промышленных роботов, в которых схват используется для удержания рабочего инструмента, относительно невелико. В большинстве случаев инструмент, нужный для выполнения технологической операции, крепится непосредственно к запястью робота, становясь его рабочим органом. Это может быть пульверизатор для окраски распылением, сварочные клещи для точечной сварки, сварочная головка для дуговой сварки, дисковый нож, дрель, фреза, отвертка, гайковерт и т. д.

В нашем случае используется схват RG2 Gripper.



Рис. 7.2. Схват RG2 Gripper

Схват RG2 предназначен для идеальной подгонки роботов UR и является гибким и простым в использовании. Обладает высокой гибкостью и простотой программирования. Захват RG2 работает без каких-либо внешних кабелей, поэтому любое движение робота можно выполнять, не беспокоясь о размещении кабеля.

Движение и сила схвата RG2 могут контролироваться пользователем. Управление выполняется непосредственно на пользовательском интерфейсе Universal Robots.

RG2 способен измерять ширину объекта и определять, захвачен объект или нет. Все сигналы обратной связи отправляются роботу и могут использоваться для совершения действия робота на основе входных данных.

Для данного рабочего органа в редакторе структуры программы имеется команда RG2 и используется для задания ширины раскрытия схвата и силы его сжатия.

Добавление команды: программа/ структура/ URCaps/ RG2.

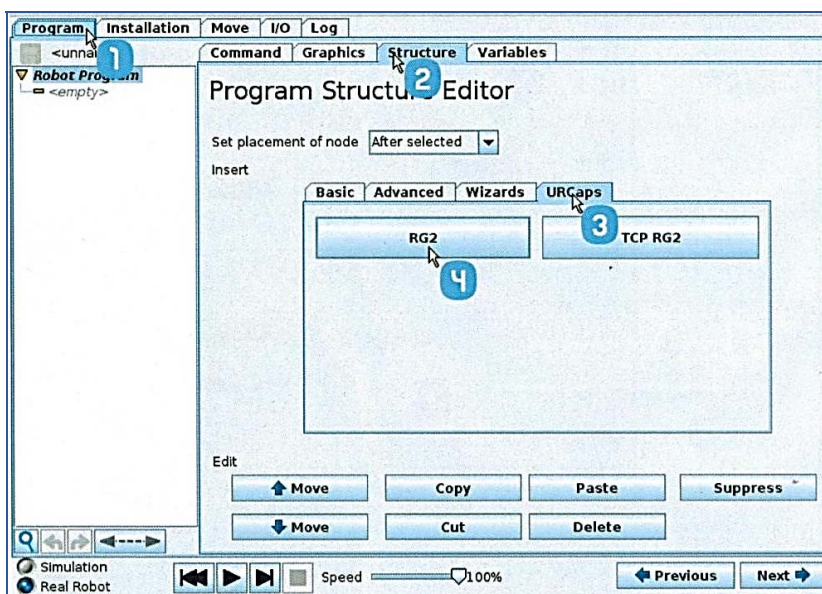


Рис. 7.3. Редактор структуры программы

В окне команды RG2 задаем нужные параметры.

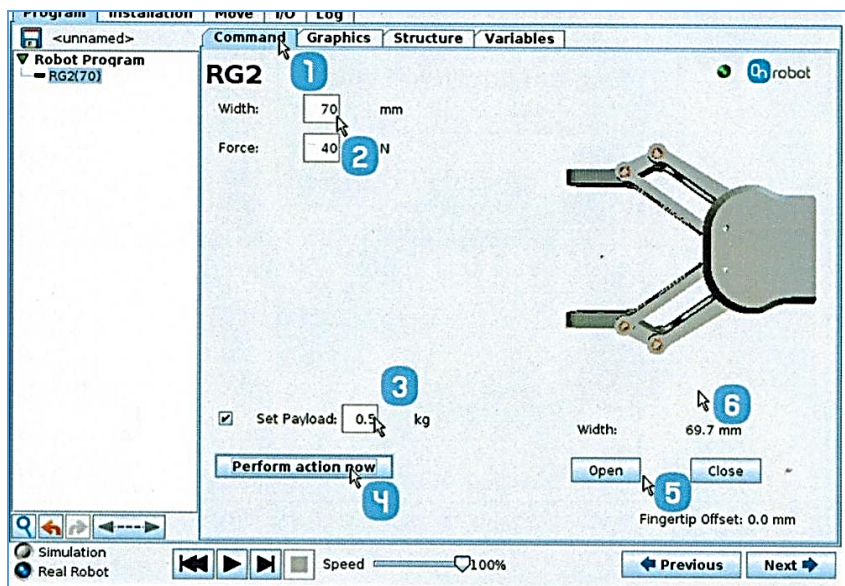


Рис. 7.4. Окно команды RG2

Задание

Запрограммировать траекторию движения робота, используя варианты из лабораторной работы № 6 и схват для переноса предметов. Для выполнения работы использовать «свободный привод».

Для управления манипулятором вручную необходимо удерживать нажатой кнопку «Свободный привод» (дублируется сзади пульта), расположенную на задней стороне подвесного пульта обучения, вручную перемещая манипулятор робота.

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Задание.
4. Листинг программы.

Лабораторная работа № 8

ИСПОЛЬЗОВАНИЕ КОМАНДЫ «PATTERN» ДЛЯ УКЛАДКИ ОДНОТИПНЫХ ДЕТАЛЕЙ

Цель работы

1. Научиться создавать шаблоны для укладки однотипных деталей.
2. Изучить команду «Паллетирование».

Основные сведения

Команда Pattern (шаблон)

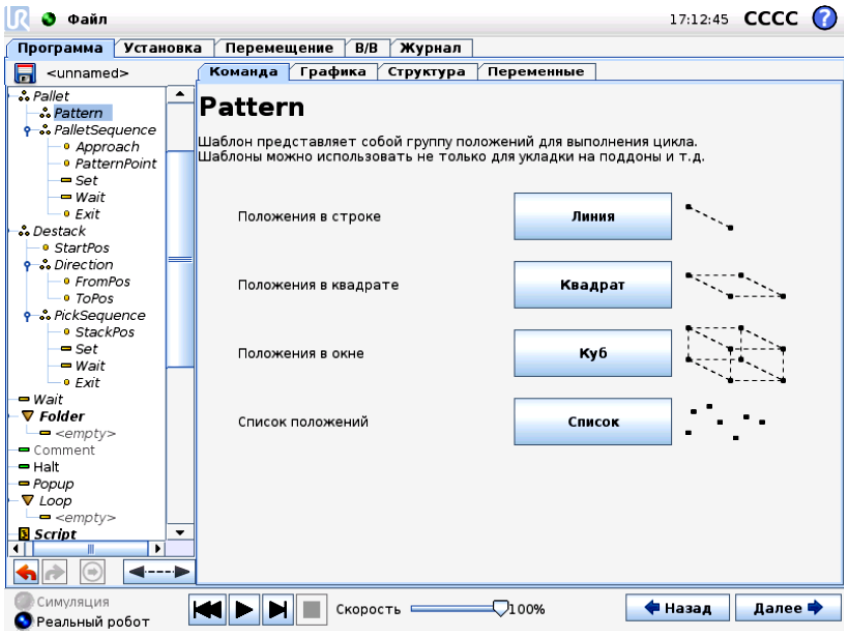


Рис. 8.1. Команда «Pattern»

Команду Pattern можно использовать для циклического перехода между положениями в программе робота. Команда Pattern соответствует одному положению при каждом выполнении.

Можно задать один из четырех типов шаблона. Первые три, «Линия», «Квадрат» или «Куб», можно использовать для положений в регулярном шаблоне.

Правильные шаблоны определяются числом характеристических точек, где эти точки определяют границы шаблона.

Для типа «Линия» это две концевые точки, для типа «Квадрат» – три из четырех угловых точек, а для типа «Куб» – это четыре из восьми угловых точек.

Программист вводит число положений по каждому краю шаблона.

Затем контроллер робота рассчитывает отдельные положения шаблона, пропорционально складывая векторы границ.

Если положения, которые требуется пройти, не укладываются в рамки правильного шаблона, можно выбрать параметр Список, в котором будет представлен список всех положений, указанных программистом.

Таким образом, можно реализовать любую компоновку положений.

Команда Pallet (платформа)

Операция Pallet позволяет выполнять последовательность движений в нескольких местах, заданных в качестве шаблона. В каждой позиции шаблона последовательность движений будет выполняться относительно позиции шаблона.

Создание программы операции «Pallet»

Выполните следующие действия:

1. Определите шаблон.
2. Создайте Последовательность платформы для подбора и размещения в каждой отдельной точке. Эта последовательность описывает, что необходимо сделать в каждой позиции шаблона.
3. Для определения контрольных точек, которые должны соответствовать позициям шаблона, используйте переключатель на экране команд последовательности.

Последовательность платформы/ Закрепляемая последовательность (Pallet Sequence/ Anchorable Sequence)

Pattern: Square

- Pattern
 - Set pattern: Square
 - Set objects between
 - Point 1 to 2
 - Point 2 to 3
 - Teach the 4 corners

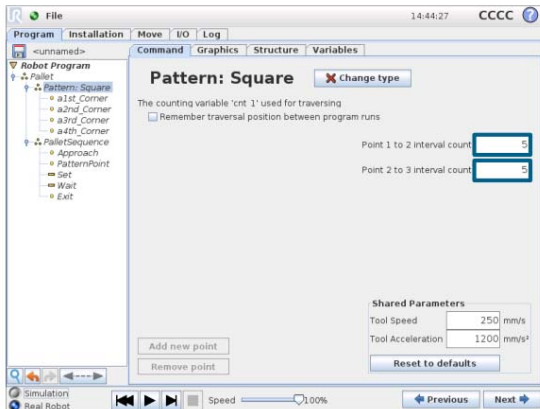
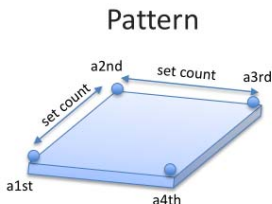


Рис. 8.2. Команда «Pattern» – квадрат

Robot Program Pallet

```
Pattern: Square
a1st_Corner
a2nd_Corner
a3rd_Corner
a4th_Corner
PalletSequence
Approach_1
PatternPoint_1
Set DO[0] =True
Wait 0.5
Exit_1
```

PalletSequence

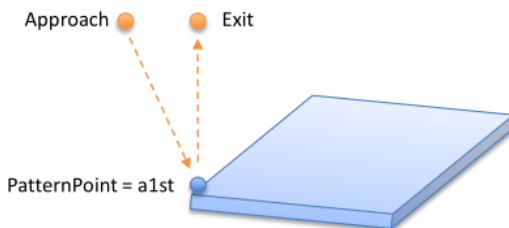


Рис. 8.3. Точки позиции входа и выхода

Точки Approach и Exit – это точки входа и выхода соответственно. В узле «Последовательность платформы» (Pallet Sequence) движения манипулятора робота выполняются относительно положения платформы.

Последовательность выполняется таким образом, что манипулятор робота оказывается в позиции, заданной шаблоном в положении якоря/ точке шаблона.

Все остальные позиции будут перемещены в соответствии с этой позицией.

Не используйте команду Переместить (Move) внутри последовательности, так как ее не удастся соотнести с положением якоря.

«Перед запуском» (BeforeStart)

Дополнительная последовательность «Перед запуском» выполняется непосредственно перед запуском операции. Ее можно использовать для ожидания готовых сигналов.

«После окончания» (AfterEnd)

Дополнительная последовательность «После окончания» выполняется после завершения операции. Ее можно использовать для отправки сигнала запуска конвейера, подготавливая место для следующей платформы.

Задание

Создать шаблон, для укладки используя теоретические сведения. Варианты заданий выдает преподаватель.

Содержание отчета

1. Титульный лист.
2. Цель работы.
3. Задание.
4. Листинг программы.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. C01UE201-1503_RA605 [Электронный ресурс] : руководство пользователя. – Режим доступа: Software manual.pdf. – Дата доступа: 18.10.2020.
2. Официальный сайт компании HIWIN [Электронный ресурс]. – Режим доступа: <https://www.hiwin.com>. – Дата доступа: 11.10.2020.
3. Официальный сайт компании Universal Robots [Электронный ресурс]. – Режим доступа: <https://www.universal-robots.com>. – Дата доступа: 12.10.2020.

Учебное издание

ОКОЛОВ Андрей Ромуальдович
МАТРУНЧИК Юлия Николаевна

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
ПРОМЫШЛЕННЫХ РОБОТОВ**

Учебно-методическое пособие
для студентов учреждений высшего образования
по специальностям 1-53 01 01 «Автоматизация технологических
процессов и производств», 1-53 01 06 «Промышленные роботы
и робототехнические комплексы»

Редактор *Е. В. Герасименко*
Компьютерная верстка *Н. А. Школьниковой*

Подписано в печать 30.03.2021. Формат 60×84 ¹/₁₆. Бумага офсетная. Ризография.
Усл. печ. л. 3,89. Уч.-изд. л. 3,05. Тираж 100. Заказ 805.

Издатель и полиграфическое исполнение: Белорусский национальный технический университет.
Свидетельство о государственной регистрации издателя, изготовителя, распространителя
печатных изданий № 1/173 от 12.02.2014. Пр. Независимости, 65. 220013, г. Минск.