

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Кузбасский государственный технический университет  
имени Т. Ф. Горбачева»

Институт информационных технологий, машиностроения и  
автотранспорта  
Кафедра информационных и автоматизированных  
производственных систем

Николай Петрович Курышкин

## **Системы управления и программирование промышленных роботов**

Методические материалы к практическим занятиям и  
самостоятельной работе

Рекомендовано учебно-методической комиссией  
направления подготовки 15.04.04 Автоматизация  
технологических процессов и производств, профиль «Авто-  
матизация и роботизация в промышленности»  
в качестве электронного издания  
для использования в учебном процессе

Кемерово 2024

Рецензенты: Любимов О. В. – – канд. тех. наук, доцент кафедры информационных и автоматизированных производственных систем ФГБОУ ВО «Кузбасский государственный технический университет имени Т. Ф. Горбачева»

Садовец В. Ю. – – канд. тех. наук, доцент кафедры информационных и автоматизированных производственных систем ФГБОУ ВО «Кузбасский государственный технический университет имени Т. Ф. Горбачева»

**Курышкин, Н.П. Системы управления и программирование промышленных роботов:** Методические материалы к практическим занятиям и самостоятельной работе для обучающихся по направлению 15.04.04 Автоматизация технологических процессов и производств, профиль «Автоматизация и роботизация в промышленности» всех форм обучения / сост. Н. П. Курышкин; Кузбасский государственный технический университет имени Т. Ф. Горбачева. – Кемерово, 2024. – Текст: электронный.

Раскрыто содержание практических занятий по темам, предусмотренным рабочей программой дисциплины «Системы управления и программирование промышленных роботов». Приведены методические материалы для проведения восьми практических занятий по программированию промышленных роботов *FANUC* и *KUKA*. Рассмотрены методики программирования роботов как в онлайн-режиме с использованием пультов *iPendant* и *smartPAD*, так и в офлайн режиме в программной среде *ROBOGUIDE*. Описание каждого практического занятия заканчивается индивидуальным заданием, контрольными вопросами и списком литературы.

© Кузбасский государственный  
технический университет имени  
Т. Ф. Горбачева, 2024

© Курышкин Н. П.,  
составление, 2024

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	5
1. КАЛИБРОВКА СВАРОЧНОГО РОБОТА КУКА.....	6
1.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ .....	6
1.2. МЕТОДИКА КАЛИБРОВКИ ИНСТРУМЕНТА TOOL.....	9
1.3. МЕТОДИКА КАЛИБРОВКИ БАЗЫ BASE.....	14
1.4. ЗАДАНИЕ .....	16
1.5. КОНТРОЛЬНЫЕ ВОПРОСЫ .....	16
1.6. ЛИТЕРАТУРА .....	16
2. НАСТРОЙКА СИСТЕМ КООРДИНАТ РОБОТА FANUC.....	17
2.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ .....	17
2.2. МЕТОДИКА НАСТРОЙКИ СИСТЕМЫ КООРДИНАТ ИНСТРУМЕНТА (TOOL FRAME) .....	19
2.3. МЕТОДИКА НАСТРОЙКИ ПОЛЬЗОВАТЕЛЬСКОЙ СИСТЕМЫ КООРДИНАТ (USER FRAME) .....	22
2.4. НАСТРОЙКА НАГРУЗКИ.....	24
2.5. ЗАДАНИЕ .....	25
2.6. КОНТРОЛЬНЫЕ ВОПРОСЫ .....	26
2.7. ЛИТЕРАТУРА .....	26
3. ПРОГРАММИРОВАНИЕ СВАРОЧНОГО РОБОТА КУКА...27	
3.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ .....	27
3.2. ЗАДАНИЕ .....	37
3.3. КОНТРОЛЬНЫЕ ВОПРОСЫ .....	37
3.4. ЛИТЕРАТУРА .....	37
4. ПРОГРАММИРОВАНИЕ ПРОМЫШЛЕННОГО РОБОТА FANUC .....	38
4.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ .....	38
4.2. ЗАДАНИЕ .....	45
4.3. КОНТРОЛЬНЫЕ ВОПРОСЫ .....	45
4.4. ЛИТЕРАТУРА .....	46

5. СОЗДАНИЕ РОБОТИЗИРОВАННОЙ ЯЧЕЙКИ В ПРОГРАММНОЙ СРЕДЕ ROBOGUIDE .....	47
5.1. КРАТКИЙ ОБЗОР ПАКЕТА ROBOGUIDE V9 .....	47
5.2. СОЗДАНИЕ РОБОТИЗИРОВАННОЙ ЯЧЕЙКИ.....	48
5.3. ЗАДАНИЕ .....	57
5.4. КОНТРОЛЬНЫЕ ВОПРОСЫ .....	59
5.5. ЛИТЕРАТУРА .....	59
6. ПРОГРАММИРОВАНИЕ РОБОТА В СРЕДЕ ROBOGUIDE .	60
6.1. МЕТОДИКА РАЗРАБОТКИ ПРОГРАММЫ .....	60
6.2. ЗАДАНИЕ .....	72
6.3. КОНТРОЛЬНЫЕ ВОПРОСЫ .....	73
6.4. ЛИТЕРАТУРА .....	73
7. РАЗРАБОТКА ПРОЕКТА СВАРОЧНОГО РТК С ДВУХОСЕВЫМ ПОЗИЦИОНЕРОМ В ПРОГРАММНОЙ СРЕДЕ ROBOGUIDE.....	74
7.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ .....	74
7.2. ЗАДАНИЕ .....	83
7.3. КОНТРОЛЬНЫЕ ВОПРОСЫ .....	84
7.4. ЛИТЕРАТУРА .....	84
8. УПРАВЛЕНИЕ ЦИФРОВЫМИ ВХОДАМИ/ВЫХОДАМИ В ПРОГРАММНОЙ СРЕДЕ ROBOGUIDE .....	85
8.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ .....	85
8.2. ЗАДАНИЕ .....	92
8.3. КОНТРОЛЬНЫЕ ВОПРОСЫ .....	92
8.4. ЛИТЕРАТУРА .....	93

## ВВЕДЕНИЕ

Методические материалы к практическим занятиям по дисциплине «Системы управления и программирование промышленных роботов» представляют собой подробные инструкции для проведения восьми практических занятий.

Занятия проводятся в специализированной лаборатории, оснащённой сварочным роботом *KUKA KR 8 R1620*, учебной роботизированной ячейкой с промышленным роботом *FANUC ER-4iA* и компьютерами с программным комплексом *ROBOGUIDE V9*.

Первые два занятия посвящены изучению ввода в эксплуатацию роботов *KUKA* и *FANUC* путём калибровки или настройки их систем координат. На третьем и четвертом занятиях обучающиеся приобретают практические навыки программирования роботов *KUKA* и *FANUC* с использованием пультов *smartPAD* и *iPendant*, соответственно.

На пятом – восьмом занятиях вырабатываются практические компетенции по разработке цифровых двойников роботизированных технологических комплексов и программированию роботов *FANUC*, входящих в их состав. Проекты выполняются в программной среде *ROBOGUIDE V9*.

Заголовок каждой из восьми глав совпадает с темой соответствующего практического занятия. Каждая глава заканчивается индивидуальным заданием, контрольными вопросами и списком литературы.

## 1. КАЛИБРОВКА СВАРОЧНОГО РОБОТА KUKA

Цель – приобретение навыков калибровки промышленного сварочного робота *KUKA*.

С этой целью необходимо выполнить калибровку систем координат инструмента и базы промышленного робота *KR 8 R1620*, оснащенного сварочным оборудованием для сварки в среде защитного газа *MIG-MAG*. Кроме этого, необходимо выполнить калибровку дополнительной нагрузки от элементов сварочного оборудования, установленного на звенья манипулятора.

### 1.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ

Программирование и функционирование промышленного робота (ПР) выполняется в 3D-пространстве, которое нормируется *правой* прямоугольной системой координат ( $x, y, z$ ). Положение и ориентация различных взаимодействующих объектов роботизированного технологического комплекса (РТК) также определяется связанными с ними правыми декартовыми системами координат.

На рис. 1.1 представлен минимальный набор систем координат, которые используются при программировании и функционировании ПР: *WORLD* (мировая, абсолютная, неподвижная), *ROBROOT* («корень», основание робота), *BASE* (система координат заготовки или детали, с которой взаимодействует ПР), *TOOL* (система координат инструмента).

Система координат *WORLD* представляет собой жестко заданную, неподвижную декартову систему координат. Она является исходной для систем координат *ROBROOT* и *BASE*. Если ПР закреплён стационарно, то система координат *WORLD* совпадает с системой *ROBROOT*.

Декартова система координат *ROBROOT* всегда расположена в центре вертикальной вращательной кинематической пары *J1*. Она описывает положение робота по отношению к системе координат *WORLD*, если он подвижный. Положение осей этой системы жестко регламентировано: ось  $x$  направлена к наблюдателю, если он стоит перед роботом лицом к нему, ось  $y$  направлена вправо от наблюдателя, а ось  $z$  – вертикально вверх.

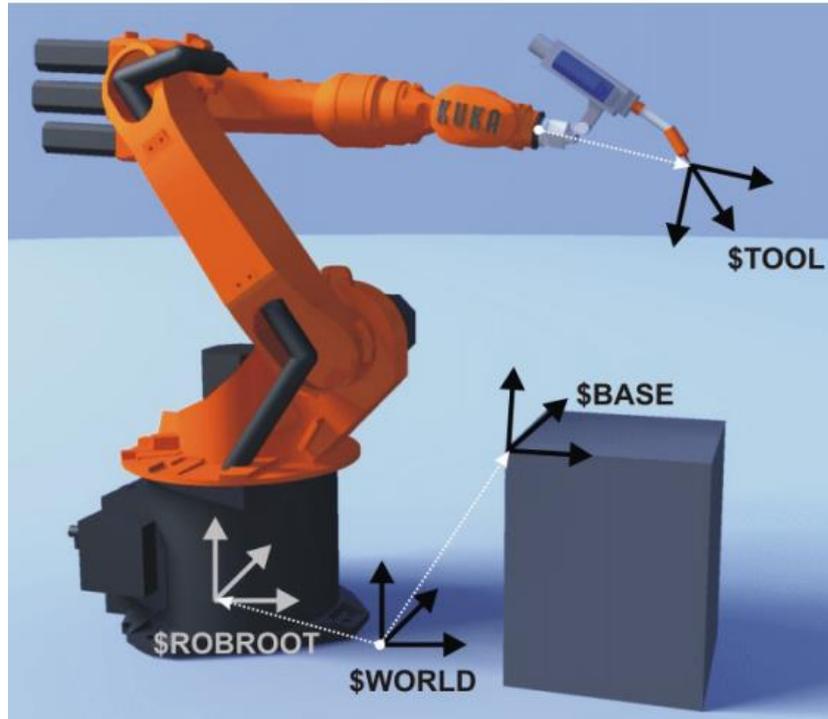


Рис. 1.1 Общий вид систем координат

Система координат *BASE* задается в системе координат *WORLD*. По умолчанию система координат *BASE* совпадает с системой координат *WORLD*.

Декартова система координат *TOOL* расположена в главной рабочей точке инструмента *TCP* (*tool center point*). По умолчанию исходной точкой системы координат *TOOL* является центр монтажного фланца. В этом случае она называется системой координат *FLANGE*.

Процесс калибровки систем координат состоит из двух этапов. На первом этапе система *FLANGE* заменяется системой *TOOL* с необходимой ориентацией осей. На втором этапе вводится система координат *BASE*. Она связывается с заготовкой. Возможна калибровка шестнадцати инструментов и тридцати двух баз.

Преимущества калибровки *TOOL*:

- инструмент можно перемещать прямолинейно в направлении удара;
- инструмент можно повернуть вокруг *TCP*, не изменяя положения *TCP*;
- в программном режиме запрограммированная скорость перемещения сохраняется вдоль траектории перемещения *TCP*.

Преимущества калибровки *BASE*:

- ТСР можно вручную перемещать по рабочей поверхности, заготовки;
  - возможно программирование точек относительно базы.
- Если необходимо переместить базу, например, вследствие перемещения рабочей поверхности, точки смещаются вместе с ней, и новое программирование не требуется.

Важным этапом подготовки ПР к эксплуатации является калибровка дополнительной нагрузки на звенья манипулятора (рис. 1.2). В качестве такой нагрузки выступает полезная нагрузка на фланце – инструмент или захватное устройство с захваченным объектом **1**, источник сварочного тока или механизм подачи материала **2**, система электроснабжения, клапаны, запас материала и т.п. **3, 4**.

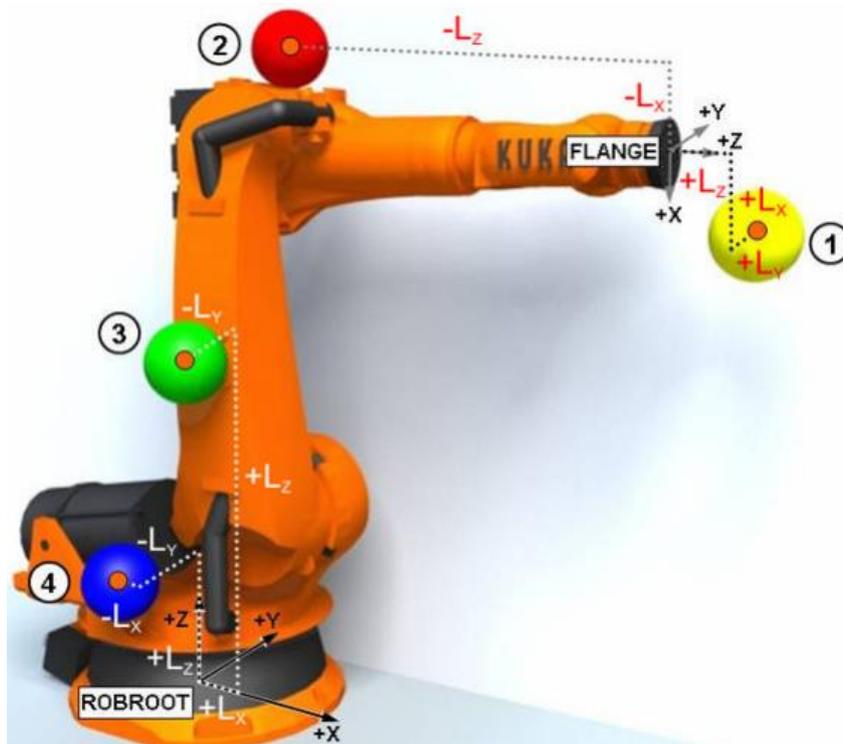


Рис. 1.2 Схема расположения дополнительной нагрузки

Введённые данные дополнительной нагрузки оказывают воздействие на большое количество процессов системы управления. К ним относятся: алгоритмы системы управления, контроль скорости и ускорения, контроль моментов и энергии и т.д. Поэтому крайне важно ввести корректные данные нагрузки. Если

робот выполняет перемещения с верно установленными данными дополнительной нагрузки, то:

- робот работает с высокой точностью и эффективностью;
- возможны операции перемещения с оптимальным временем цикла;
- повышается срок службы робота (благодаря малому износу элементов приводов).

## 1.2. МЕТОДИКА КАЛИБРОВКИ ИНСТРУМЕНТА TOOL

Калибровка инструмента означает, что создаётся система координат, начало которой находится в точке *TCP* (рис. 1.3). При калибровке сохраняются координаты точки *TCP* в системе координат фланца, а также поворот осей *X*, *Y*, *Z* относительно друг друга (параметры *A*, *B*, *C*).

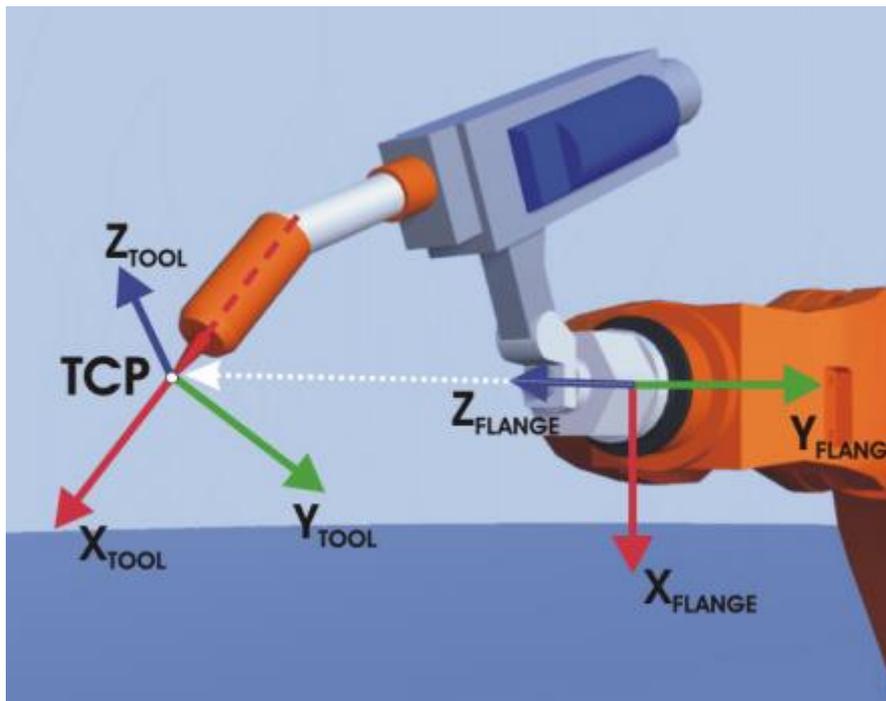


Рис. 1.3 Результат калибровки инструмента

Калибровка инструмента выполняется с помощью выносного пульта управления роботом *smartPAD*. Необходим также калибровочный конус в виде диска с заострённым вертикальным ко-

ническим стержнем, установленном на рабочем столе в зоне досягаемости робота.

Калибровка инструмента состоит из трёх шагов:

Шаг	Описание
1	<p>Определение положения точки <i>TCP</i>.</p> <p>На выбор предлагаются два метода:</p> <ul style="list-style-type: none"> <li>– XYZ, 4 точки;</li> <li>– база отсчёта XYZ.</li> </ul>
2	<p>Определение ориентации системы координат <i>TOOL</i></p> <p>На выбор предлагаются следующие методы:</p> <ul style="list-style-type: none"> <li>– ABC, универсальный;</li> <li>– ABC, две точки.</li> </ul>
3	<p>Ввод данных нагрузки инструмента:</p> <ul style="list-style-type: none"> <li>– масса инструмента или схвата с деталью, кг;</li> <li>– координаты центра масс в системе координат FLANGE, мм;</li> <li>– угловое положение главных центральных осей инерции инструмента (схвата с деталью) относительно осей FLANGE, град.;</li> <li>– моменты инерции инструмента (схвата с деталью) относительно его главных центральных осей инерции, кг·м<sup>2</sup>.</li> </ul>

Калибровка инструмента начинается с открытия на экране *smartPAD* главного меню. Далее выбираем «Ввод в эксплуатацию» > «Управление инструментами и базами». Далее необходимо коснуться клавиши «Инструмент/Заготовка» и нажать клавишу «Добавить» (рис. 1.4).

В открывшемся окне **1** (рис. 1.5) выбрать «свободный» номер инструмента, задать имя нового инструмента в поле ввода и выбрать вид калибруемого объекта. Из двух вариантов всплывающего меню «Инструмент» или «База» выбрать «Инструмент».

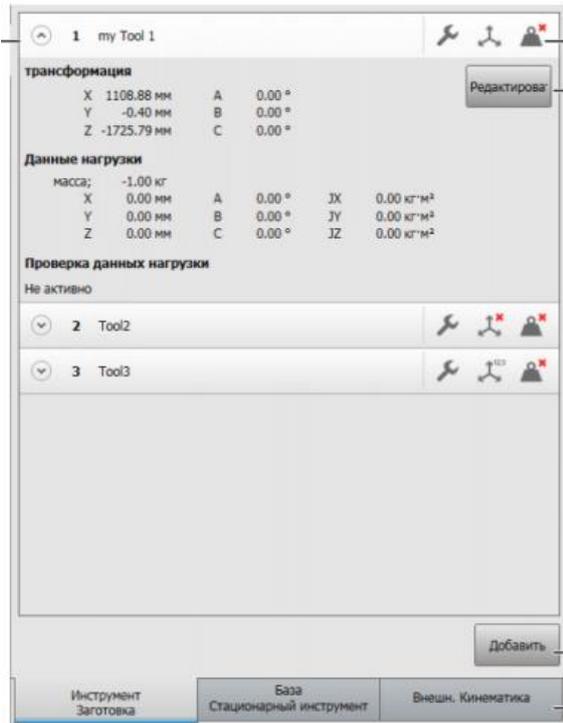


Рис. 1.4 Добавление инструмента

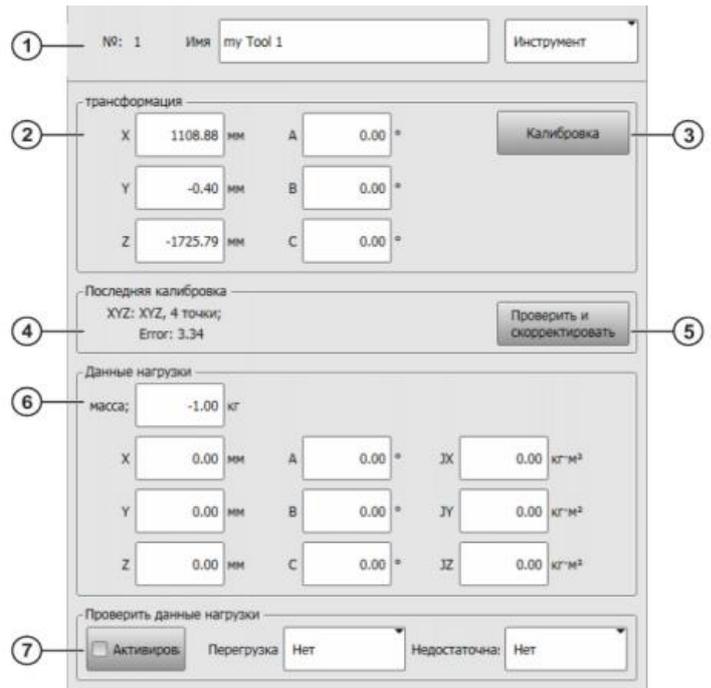


Рис. 1.5 Создание инструмента

В поле 2 «трансформация» коснуться клавиши 3 «Калибровка» и в контекстном меню выбрать метод калибровки *XYZ, 4 точки*. С помощью этого метода выполняется **первый шаг** калибровки. После его выполнения в этом поле отобразится результат калибровки – координаты точки *TCP* в системе координат *FLANGE*. Ориентация осей *TOOL* (*A, B, C*) отобразится после выполнения второго шага.

Поле 4 используется в случае, если калибровка инструмента с заданным номером была выполнена ранее, и её необходимо проверить или скорректировать прикосновением к клавише 5. Поле 6 заполняется на третьем шаге калибровки.

#### **Калибровка методом «XYZ, 4 точки».**

После выбора метода калибровки на экране *smartPAD* появится изображение рисунка 1.6 с отсутствующими данными трансформации. Вместо них будет запись «*Выполнить Touch Up*». После прикосновения к первой строке она активируется синим цветом. Далее, используя кнопки ручного управления или 6D-мышь, перевести предполагаемую точку *TCP* к острию калибровочного конуса и коснуться клавиши «Корректировка». Первая точка запишется. Подоб-

ным образом необходимо записать ещё три точки, подходя к острию конуса с разных сторон (рис. 1.7).

Точка калибровки		трансформация				
 Точка измерения 1	X	1765.00 мм	Y	0.00 мм	Z	1784.00 мм
	A	0.00 °	B	90.00 °	C	0.00 °
 Точка измерения 2	X	1764.70 мм	Y	-32.29 мм	Z	1784.00 мм
	A	0.00 °	B	90.00 °	C	1.05 °
 Точка измерения 3	X	1754.39 мм	Y	-32.10 мм	Z	1797.03 мм
	A	-1.05 °	B	89.47 °	C	0.00 °
 Точка измерения 4	X	1754.09 мм	Y	-45.91 мм	Z	1797.03 мм
	A	-1.50 °	B	89.47 °	C	0.00 °

**Результат калибровки**

X	1108.88 мм	A	0.00 °
Y	-0.40 мм	B	0.00 °
Z	-1725.79 мм	C	0.00 °

Ошибка измерения: 3.34

Рис. 1.6 Калибровка методом «XYZ, 4 точки»

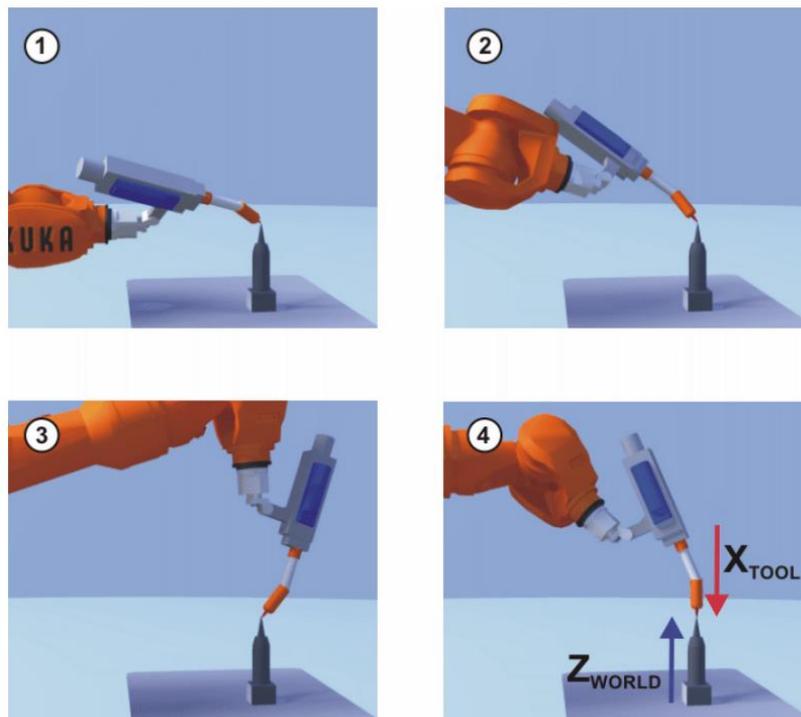


Рис. 1.7 Метод 4 точек

Обратим внимание на калибровку четвёртой точки (рис. 1.7). Положение инструмента может быть произвольным, однако сварочная горелка в данном случае расположена строго вертикально. Такое положение инструмента не случайное. Оно будет использовано при выполнении второго шага калибровки.

По окончании калибровки её результат высветится в нижней части экрана (рис. 1.6) с ошибкой в миллиметрах. Допускается ошибка на более 1,5 мм. Таким образом, первый шаг калибровки считается выполненным. Нажатием клавиши «Сохранить» результат сохраняется. В случае если ошибка превышает допустимый предел, калибровку по точкам следует скорректировать.

#### **Калибровка методом «АВС, унив.».**

**Второй шаг** калибровки позволяет сориентировать оси системы координат инструмента, как показано на рис. 1.3. Главным условием калибровки инструмента в виде сварочной горелки, чтобы ось  $X_{TOOL}$  была бы направлена по линии «удара» (совпадала с направлением выхода сварочной проволоки). Направление других осей для данного вида инструмента безразлично. С учётом этого условия во вкладке «Управление инструментами и базами», выбрав калибруемый инструмент, касаемся клавиши «Калибровка» и выбираем метод «АВС, унив.». В выпавшей вкладке выбираем способ 5D и используем то положение инструмента, как показано для точки 4 на рис. 1.7. Последовательным нажатием клавиш «Корректировка» и «Сохранить» записываем результат второго шага калибровки.

**Третий шаг** калибровки инструмента выполняется путём ввода данных нагрузки в поле **6** во вкладке «Управление инструментами и базами» с выбором соответствующего инструмента (рис. 1.5). В случае если в контроллере робота имеется опция *KU-КА.LoadDetermination*, то данные нагрузки определяются автоматически.

Данные дополнительной нагрузки от механизма подачи сварочной проволоки на ось  $J3$  (нагрузка **2** на рис. 1.2) вводятся по следующему алгоритму. В главном меню выбрать «Ввод в эксплуатацию» > «Данные дополнительной нагрузки». Далее ввести номер оси, на которой закреплена дополнительная нагрузка. Нажать кнопку «Далее» для подтверждения. Затем ввести данные нагрузки и нажать кнопку «Далее» для подтверждения. В завершении нажать кнопку «Сохранить».

Данные нагрузки можно взять из следующих источников:

- технический паспорт на соответствующий инструмент или механизм;

- САD-модель инструмента или механизма, спроектированного в какой-либо среде САПР;
- расчёт вручную.

Проверка правильности и корректности калибровки инструмента выполняется путём его ручного перемещения. Для этого на пульте *smartPAD* необходимо переключить ручное управление в систему координат откалиброванного инструмента и нажимать кнопки +, – оси  $X$ . Инструмент должен двигаться строго вдоль оси удара (оси  $X_{TOOL}$ ). После этого, нажимая кнопки +, –  $A, B, C$ , убедиться, что инструмент вращается вокруг *неподвижной* точки  $TCP$ .

### 1.3. МЕТОДИКА КАЛИБРОВКИ БАЗЫ BASE

Калибровка базы означает создание декартовой системы координат в определённом месте окружения робота (рис. 1.1). При калибровке базы сначала задаётся положение начала её системы координат, а затем – направление осей  $X$  и  $Y$ . Калибровку базы выполняют уже откалиброванным инструментом.

Калибровка базы может выполняться двумя методами: «три точки» и «косвенно». Вторым методом используется редко и тогда, когда начало системы координат базы находится вне зоны его досягаемости точкой  $TCP$  инструмента (в теле заготовки или за пределами рабочего пространства робота).

**Метод «три точки»** является основным и предполагает запись координат точки, являющейся началом системы координат базы **1**, точки на положительном направлении оси  $X_{BASE}$  **2** и точки на плоскости  $X_{BASE} Y_{BASE}$  **3** (рис. 1.8). Последовательность калибровки следующая. Сначала выбираем в главном меню «Ввод в эксплуатацию» > «Управление инструментами и базами». Далее необходимо коснуться клавиши «База/Стационарный инструмент» и нажать клавишу «Добавить» (рис. 1.4). Далее в открывшемся окне, похожем на окно калибровки инструмента без поля «Данные нагрузки» (рис. 1.5) выбрать «свободный» номер базы, присвоить ей имя и в поле «База/Стационарный инструмент» выбрать «База». Далее нажать клавишу «Калибровка» и выбрать из двух методов метод «3 точки».

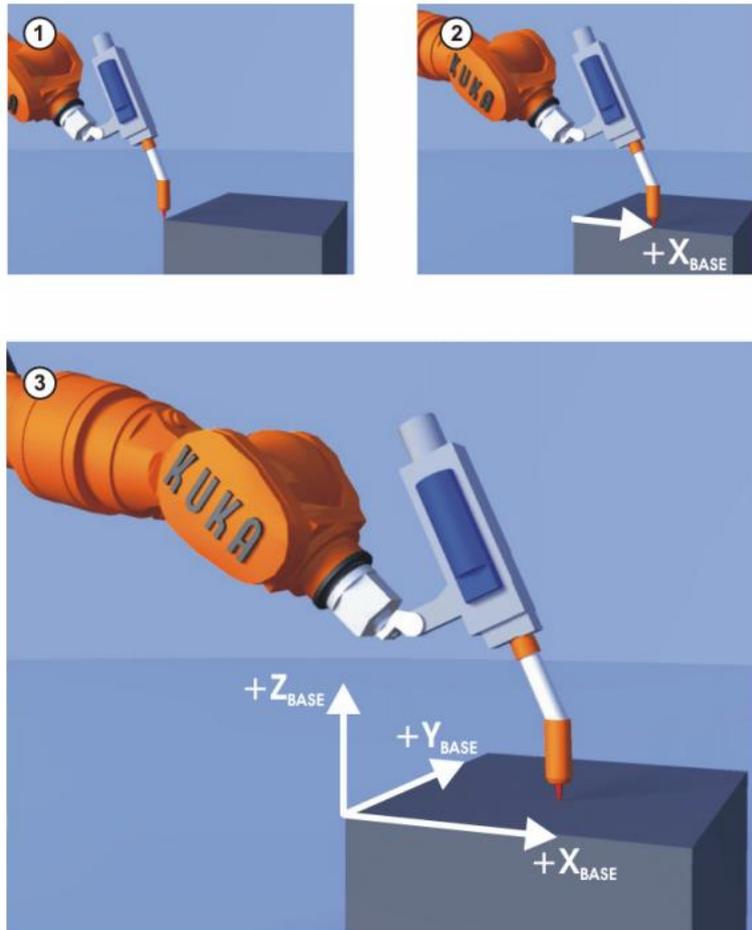


Рис. 1.8 Метод «3 точки» для калибровки базы

Используя кнопки ручного управления или 6D-мышь в системе координат *WORLD*, выполнить подвод точки *TCP* инструмента к началу системы координат новой базы. В окне, похожем на окно калибровки инструмента (рис. 1.6), выбрать в раскрывающемся меню эталонный инструмент, который используется для калибровки базы. Далее коснуться строки «Начало» (она окрасится в синий цвет) и нажать клавишу «Корректировка». Рассчитанные значения отобразятся в поле «трансформация». Подобным образом выполнить запись точки на положительном направлении оси  $X_{BASE}$  и точки на плоскости  $XY$  с положительным значением  $Y$ . Расстояние между точками – (100...200) мм. Точки не должны лежать на одной прямой. После записи координат трёх точек в окне появится результат калибровки и станет доступной клавиша «Сохранить». Её нужно нажать и данные новой базы будут сохранены.

Проверка правильности калибровки базы выполняется путём перемещения точки *TCP* инструмента нажатием клавиш ручного управления  $+, - X, Y, Z$ . Точка *TCP* должна перемещаться строго па-

раллельно соответствующим осям базы. Предварительно необходимо выбрать режим ручного управления в системе координат новой базы.

#### 1.4. ЗАДАНИЕ

Выполнить калибровку сварочной горелки, установленной на фланце промышленного робота *KR 8 R1620*. Ввести данные полезной и дополнительной нагрузки в контроллер ПР. Данные нагрузки оценить самостоятельно. Выполнить также калибровку базы, связанной с объектом сварки, который предложит преподаватель.

Продемонстрировать преподавателю проверку результатов калибровки в ручном режиме.

#### 1.5. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое калибровка инструмента и базы?
2. Зачем необходимо калибровать инструмент?
3. В чём преимущество калиброванной базы?
4. Какие системы координат используются при калибровке инструмента и базы?
5. Как проверяется правильность калибровки инструмента и базы?
6. В чём заключается необходимость введения данных полезной и дополнительной нагрузки?

#### 1.6. ЛИТЕРАТУРА

1. *KUKA System Software 8.3 RUS* Инструкция по эксплуатации и программированию для конечного пользователя, режим доступа: *studylib.ru* (дата обращения 24.02.2024г.).

2. *KUKA industrial Robot*, режим доступа: <http://www.kuka-industries.com/de/career/> (дата обращения 24.02.2024г.).

## 2. НАСТРОЙКА СИСТЕМ КООРДИНАТ РОБОТА FANUC

Цель – приобретение навыков настройки систем координат промышленного робота *FANUC*.

С этой целью необходимо выполнить настройку системы координат инструмента и системы координат пользователя промышленного робота *FANUC ER-4iA*, входящего в состав учебной роботизированной ячейки. Кроме этого, необходимо ввести данные полезной нагрузки от двухпальцевого захватного устройства Schunk, смонтированного на фланце манипулятора.

### 2.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ

Программирование и функционирование промышленного робота (ПР) выполняется в 3D-пространстве, которое нормируется *правой* декартовой системой координат ( $x, y, z$ ). Положение и ориентация различных взаимодействующих объектов роботизированного технологического комплекса (РТК) также определяется связанными с ними правыми декартовыми системами координат.

При программировании и функционировании ПР *FANUC* используются четыре вида систем координат (рис. 2.1). Система обобщённых координат ( $J_1, J_2, J_3, J_4, J_5$  и  $J_6$ ) не является декартовой системой и используется редко при программировании робота методом обучения. В остальных системах координат принята единая цветовая гамма обозначения осей: ось  $X$  – красная, ось  $Y$  – зелёная и ось  $Z$  – синяя.

Мировая система координат *World* неподвижна, её начало находится в центре первой оси робота  $J_1$ . Положение осей этой системы жестко регламентировано: ось  $X$  направлена к наблюдателю, если он стоит перед роботом лицом к нему, ось  $Y$  направлена вправо от наблюдателя, а ось  $Z$  – вертикально вверх.

Система координат инструмента (*Tool Frame*) связывается с инструментом, установленном на фланце шестой оси ПР  $J_6$ . Если инструмент не установлен или система координат инструмента не настроена, то последняя располагается на этом фланце, как показано на рис. 2.1. Начало системы координат инструмента – это главная точка инструмента *TCP* (*Tool Center Point*). При настройке системы координат инструмента точка *TCP* переносится

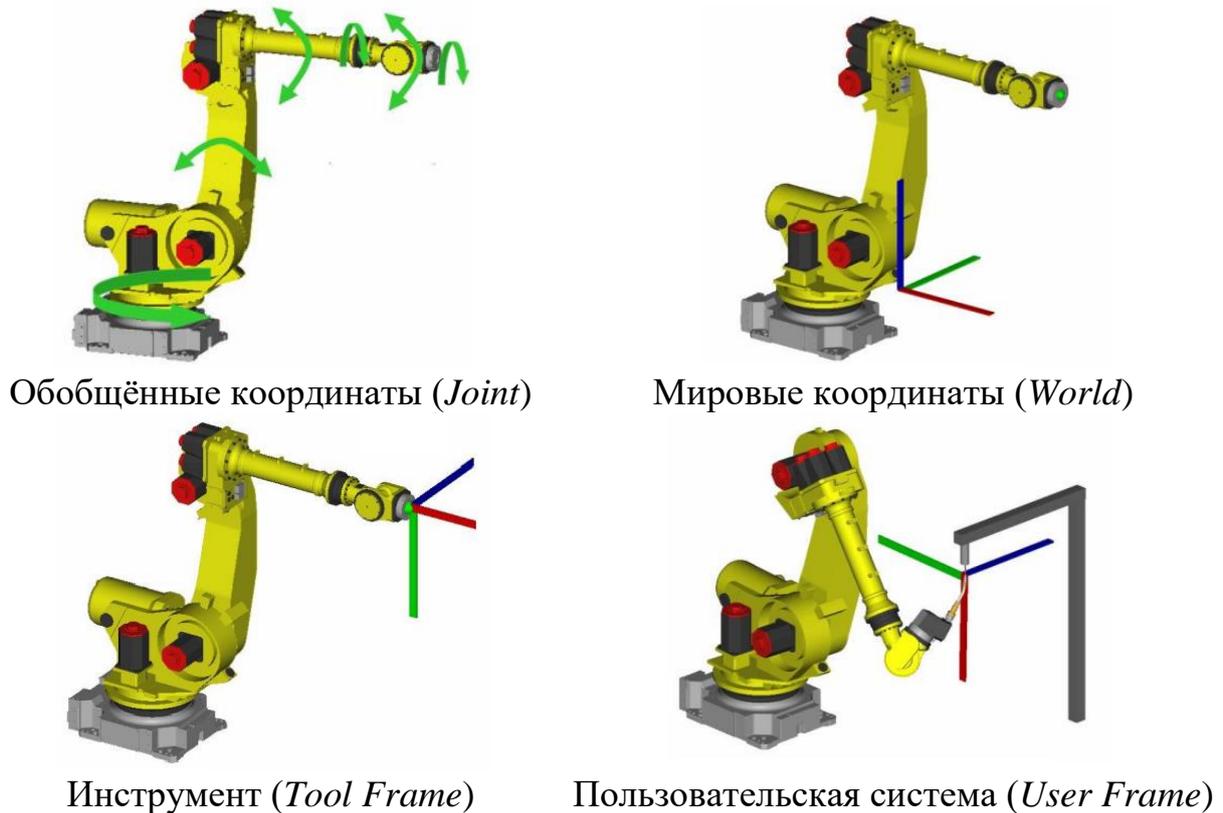


Рис. 2.1 Системы координат робота *FANUC*

в реальную главную точку инструмента. Такой точкой является, например, конец проволоки сварочной горелки, конец выходного отверстия сопла краскопульта, середина расстояния между подвижными губками захватного устройства и т.п.

Пользовательская система координат (*User Frame*) по умолчанию совпадает с мировой системой. Настройка этой системы заключается в её переносе на объект, с которым будет взаимодействовать робот. Настройка *User Frame* выполняется с помощью уже настроенного инструмента.

Возможна настройка десяти инструментов.

Преимущества настройки системы координат инструмента (*Tool Frame*):

- инструмент можно перемещать прямолинейно в направлении удара;
- инструмент можно повернуть вокруг ТСР, не изменяя положения ТСР;
- в программном режиме запрограммированная скорость перемещения сохраняется вдоль траектории перемещения ТСР.

Преимущества настройки пользовательской системы координат (*User Frame*):

- ТСП можно вручную перемещать по рабочей поверхности, заготовки;
- возможно программирование точек относительно *User Frame*. Если необходимо переместить *User Frame*, например, вследствие перемещения рабочей поверхности, точки смещаются вместе с ней, и новое программирование не требуется.

Важным этапом подготовки ПР к эксплуатации является введение полезной и дополнительной нагрузки на звенья манипулятора. В качестве такой нагрузки выступает полезная нагрузка на фланце – инструмент или захватное устройство с захваченным объектом. Источник сварочного тока или механизм подачи материала, установленные на оси *J3*, создают дополнительную нагрузку.

Введённые данные полезной и дополнительной нагрузки оказывают воздействие на большое количество процессов системы управления. К ним относятся: алгоритмы системы управления, контроль скорости и ускорения, контроль моментов и энергии и т.д. Поэтому крайне важно ввести корректные данные нагрузки. Если робот выполняет перемещения с верно установленными данными дополнительной нагрузки, то:

- робот работает с высокой точностью и эффективностью;
- возможны операции перемещения с оптимальным временем цикла;
- повышается срок службы робота (благодаря малому износу элементов приводов).

## **2.2. МЕТОДИКА НАСТРОЙКИ СИСТЕМЫ КООРДИНАТ ИНСТРУМЕНТА (TOOL FRAME)**

Как известно, исходное перед настройкой положение системы координат инструмента (*Tool Frame*) – центр фланца, на который крепится инструмент. Начало этой системы координат – точка *TSP*. Направление осей жёстко определено: если смотреть на монтажную поверхность вертикально расположенного фланца, то ось *Z* будет направлена на нас, ось *Y* – влево, а ось *X* – вверх.

Для настройки *Tool Frame* необходим настроечный конус, представляющий собой диск с заострённым центральным стержнем. Этот конус ставится в произвольном месте перед роботом на поверхность рабочего стола. Настройку *Tool Frame* выполняют по методу шести точек (*Six Point XZ*). При этом первые три точки отвечают за перенос *TCP* в реальную главную точку инструмента, а последние три точки – за настройку ориентации осей *Tool Frame*.

### Метод шести точек (*Six Point XZ*).

На пульте дистанционного управления роботом *iPendant* нажать клавишу **MENU**, и выбрать пункт **SETUP** → **Frames** (рис. 2.2). После чего на экране отобразится панель настройки систем координат (рис. 2.3).

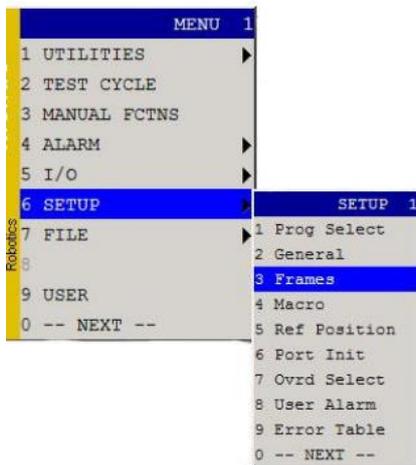


Рис. 2.2 Меню и подменю

The screenshot shows the 'SETUP Frames' panel with the following data:

Tool Frame Setup/ Direct Entry 1/10				
	X	Y	Z	Comment
1:	135.0	395.4	100.7	GRIPPER
2:	0.0	0.0	0.0	Eoat2
3:	0.0	0.0	0.0	Eoat3
4:	0.0	0.0	0.0	Eoat4
5:	0.0	0.0	0.0	Eoat5
6:	0.0	0.0	0.0	Eoat6
7:	0.0	0.0	0.0	Eoat7
8:	0.0	0.0	0.0	Eoat8
9:	0.0	0.0	0.0	Eoat9
10:	0.0	0.0	0.0	Eoat10

Active TOOL \$MNUTOOLNUM[1] = 1

Buttons at the bottom: [ TYPE ] DETAIL [ OTHER ] CLEAR SETIND

Рис. 2.3 Панель настройки *Tool Frame*

Каждая сенсорная клавиша нижней строки панели настройки дублируется физической клавишей *F1*, ... *F5*. Поэтому далее, выбрав предварительно курсором номер настраиваемого инструмента, нажимаем клавишу *F3* [*OTHER*] и выбираем пункт *Tool Frame*. После этого изменится перечень сенсорных клавиш в нижнем ряду. Надо нажать клавишу *F2* [*METHOD*] и выбрать *Six Point XZ*. После выбора метода настройки появится меню настройки (рис. 2.4). В этом окне после настройки системы координат инструмента под номером 1 появятся координаты точки *TCP* в системе координат фланца и поворот в градусах каждой из осей новой системы относительно той же системы координат

фланца:  $W$  – оси  $X$ ,  $P$  – оси  $Y$  и  $R$  – оси  $Z$ . Строку комментариев можно заполнить именем инструмента.

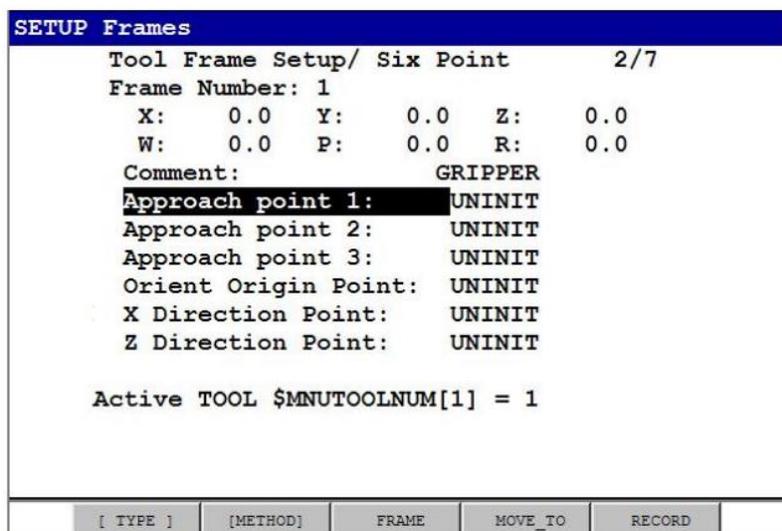


Рис. 2.4 Меню настройки *Tool Frame*

Для настройки *Tool Frame* необходимо курсором активировать строку *Approach point 1*, как показано на рис. 2.4. Далее, используя кнопки ручного управления пульта *iPendant*, в системе координат *World* подвести предполагаемую точку *TCP* к вершине настроечного конуса (рис. 2.5, а). Записать полученную точку, нажав одновременно *SHIFT* + *F5* [*RECORD*]. Состояние точки изменится с *UNINIT* (не инициализировано) на *RECORDED* (записано).

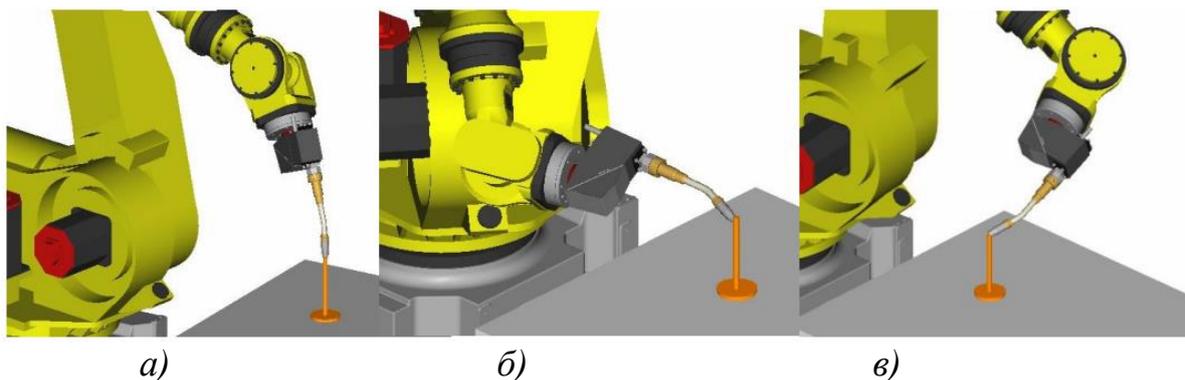


Рис. 2.5 Настройка точки *TCP*

Подобным образом записываются точки *point 2* (рис. 2.5, б) и *point 3* (рис. 2.5, в). Для более точной настройки первых трёх точек рекомендуется соблюсти два условия:

- угол между линиями инструмента в горизонтальной плоскости должен составлять примерно  $120^\circ$ ;

– угол между линией инструмента и вертикалью должен составлять примерно  $45^\circ$ .

Для записи точки *Orient Origin Point* (сориентировать исходную точку – начало системы координат) необходимо подвести к вершине настроечного конуса предполагаемую точку *TCP* инструмента в строго вертикальном положении его оси и записать эту точку.

Для записи точки *X Direction Point* (точка направления *X*) необходимо в системе координат *World* переместить точку *TCP* по оси  $+X$  на (150 ... 200) мм (см. рис. 1). Записать эту точку.

Вернуться в начало нажатием кнопки  $-X$  на те же (150 ... 200) мм. Активировать курсором точку *Z Direction Point* (точка направления *Z*) и поднять точку *TCP* нажатием кнопки  $+Z$  на (150 ... 200) мм. Записать эту точку.

После записи последней точки в меню настройки (рис. 2.4) появятся её результаты – координаты точки *TCP* (*X*, *Y*, *Z*) и ориентация этой системы (*W*, *P*, *R*) в системе координат фланца. Система координат *Tool Frame* была сориентирована так, как принято для сварочных роботов *FANUC*, а именно – при вертикальном положении сварочного инструмента оси *Tool Frame* взаимно параллельны осям мировой системы координат *World*. Это значит, что по оси удара будет направлена полуось – *Z*. Если робот не сварочный, то оси *Tool Frame* ориентируют по предпочтению, изменив при настройке положение точек *X Direction Point* и *Z Direction Point*.

Проверка правильности настройки *Tool Frame* выполняется путём ручного перемещения инструмента. Для этого на пульте *iPendant* необходимо переключить ручное управление в систему координат настроенного инструмента и нажимать кнопки  $+$ ,  $-$  оси *Z*. Инструмент должен двигаться строго вдоль оси удара (оси  $X_{TOOL}$ ). После этого, нажимая кнопки  $+$ ,  $-W$ ,  $P$ ,  $R$ , убедиться, что инструмент вращается вокруг *неподвижной* точки *TCP*.

### 2.3. МЕТОДИКА НАСТРОЙКИ ПОЛЬЗОВАТЕЛЬСКОЙ СИСТЕМЫ КООРДИНАТ (USER FRAME)

Настройка пользовательской системы координат выполняется преимущественно методом **трёх точек** (*Three Point*). Для этого, также как и для настройки *Tool Frame*, необходимо нажать кла-

вишу *MENU* и выбрать пункт *Frames* (рис. 2.2). В появившемся окне нажать функциональную клавишу *F3 [OTHER]* и выбрать *User Frame*. Далее выбрать пользовательскую систему координат, которую необходимо настроить (рис. 2.6) и нажать *F2 [DETAIL]*.

Выбрать метод настройки по трём точкам (*Three Point*), нажав функциональную клавишу *F2 [METHOD]*. Пункты *Orient Origin Point*, *X Direction Point* и *Y Direction Point* определяют положение начала *User Frame* и ориентацию её осей (рис. 2.7).

SETUP Frames				
User Frame	Setup/	Direct	Entry	1/9
	X	Y	Z	Comment
1:	0.0	0.0	0.0	UFrame1
2:	0.0	0.0	0.0	UFrame2
3:	0.0	0.0	0.0	UFrame3
4:	0.0	0.0	0.0	UFrame4
5:	0.0	0.0	0.0	UFrame5
6:	0.0	0.0	0.0	UFrame6
7:	0.0	0.0	0.0	UFrame7
8:	0.0	0.0	0.0	UFrame8
9:	0.0	0.0	0.0	UFrame9

Active UFRAME \$MNUFRAMENUM[1] = 0

[ TYPE ]    [DETAIL]    [OTHER ]    [CLEAR]    [SETIND]    >

Рис. 2.6 Панель настройки *User Frame*

Настройка *User Frame* выполняется также как и *Tool Frame*. Для этого необходимо, активировав курсором строку *Orient Origin Point*, подвести *TCP* настроенного инструмента к точке начала системы координат *User Frame* внешнего объекта и записать точку, нажав одновременно *SHIFT + F5 [RECORD]*.

SETUP Frames				
User Frame	Setup/	Three	Point	2/4
Frame Number: 1				
X:	0.0	Y:	0.0	Z: 0.0
W:	0.0	P:	0.0	R: 0.0
Comment:		UFrame1		
<b>Orient Origin Point:</b>		UNINIT		
X Direction Point:		UNINIT		
Y Direction Point:		UNINIT		

Active UFRAME \$MNUFRAMENUM[1] = 0

[ TYPE ]    [METHOD]    [FRAME]    [MOVE\_TO]    [RECORD]

Рис. 2.7 Меню настройки *User Frame*

Аналогично записываются точки на осях  $X$  и  $Y$  системы координат пользователя. Таким образом, система координат пользователя *User Frame* считается настроенной.

Проверка правильности настройки *User Frame* выполняется путём перемещения точки *TCP* инструмента нажатием клавиш ручного управления  $+,- X, Y, Z$ . Точка *TCP* должна перемещаться строго параллельно соответствующим осям *User Frame*. Предварительно необходимо активировать

Tool (.=10)	10
Jog	1
User	0
Group	1

Рис. 2.8 Выбор *User Frame*

созданную пользовательскую систему координат. Для этого надо одновременно нажать кнопки *SHIFT* + *COORD*. В появившемся окне (рис 2.8) необходимо набрать номер пользовательской системы *User Frame*, используя цифровую клавиатуру.

## 2.4. НАСТРОЙКА НАГРУЗКИ

Порядок настройки полезной нагрузки: клавишей *MENU* открыть окно меню, выбрать *NEXT* → *SYSTEM* → *Motion* (рис. 2.9). Функциональной клавишей *F3* [*DETAIL*] выбрать из списка нужный набор данных нагрузки (рис. 2.10).

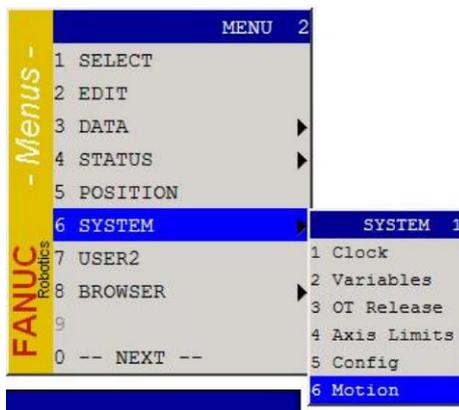


Рис. 2.9 Меню и подменю

MOTION PERFORMANCE		
Group1		1/10
No.	PAYLOAD [kg]	Comment
1	165.00	[GRIPPER ]
2	165.00	[ ]
3	165.00	[ ]
4	165.00	[ ]
5	165.00	[ ]
6	165.00	[ ]
7	165.00	[ ]
8	165.00	[ ]
9	165.00	[ ]
10	165.00	[ ]
Active PAYLOAD number = 1		

{ TYPE }    GROUP    DETAIL    ARMLoad    SETIND    >

Рис. 2.10 Панель настройки нагрузки

Параметры полезной нагрузки состоят из следующих пунктов (рис. 2.11): *Schedule No*[1] – порядковый номер нагрузки; *Payload* [kg] – масса нагрузки, кг; *Payload Center X, Y, Z* [cm] – координаты центра масс нагрузки в системе координат фланца, см;

*Payload Inertia X, Y, Z [kgfcm<sup>2</sup>]* – моменты инерции нагрузки относительно её главных центральных осей инерции, кг·см<sup>2</sup>.

MOTION/PAYLOAD SET				1/8
Group 1				
1	Schedule No[ 1 ]:	[GRIPPER]		
2	PAYLOAD	[kg]	165.00	
3	PAYLOAD CENTER X	[cm]	-28.53	
4	PAYLOAD CENTER Y	[cm]	0.00	
5	PAYLOAD CENTER Z	[cm]	27.78	
6	PAYLOAD INERTIA X	[kgfcm <sup>2</sup> ]	56.84	
7	PAYLOAD INERTIA Y	[kgfcm <sup>2</sup> ]	59.39	
8	PAYLOAD INERTIA Z	[kgfcm <sup>2</sup> ]	15.10	

Рис. 2.11 Данные полезной нагрузки

Данные дополнительной нагрузки, например, от механизма подачи сварочной проволоки на ось *Z* вводятся по следующему алгоритму. Находясь в основном меню настройки нагрузки (рис. 2.10), надо нажать функциональную клавишу *F4* [*ARMLOAD*] и ввести в соответствующей строке массу дополнительной нагрузки.

Данные нагрузки можно взять из следующих источников:

- технический паспорт на соответствующий инструмент или механизм;
- САD-модель инструмента или механизма, спроектированного в какой-либо среде САПР;
- расчёт вручную.

Для применения изменений нагрузки требуется перезагрузка системы.

## 2.5. ЗАДАНИЕ

Выполнить настройку системы координат инструмента *Tool Frame* в виде захватного устройства, установленного на фланце промышленного робота *FANUC ER-4iA*. Ввести данные полезной и дополнительной нагрузки в контроллер ПР. Данные нагрузки взять у преподавателя. Выполнить также настройку пользовательской системы координат *User Frame*, связанной с объектом, который предложит преподаватель.

Продемонстрировать преподавателю проверку результатов настройки систем координат в ручном режиме.

## 2.6. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое настройка *Tool Frame* и *User Frame*?
2. Зачем необходимо настраивать систему координат инструмента?
3. В чём преимущество настроенной системы координат пользователя?
4. Какие системы координат используются при настройке инструмента и заготовки?
5. Как проверяется правильность настройки инструмента и заготовки?
6. В чём заключается необходимость введения данных полезной и дополнительной нагрузки?

## 2.7. ЛИТЕРАТУРА

1. Официальный сайт компании *FANUC*, режим доступа [www.fanuc.eu/ru/ru](http://www.fanuc.eu/ru/ru) . (дата обращения 24.02.2024г.).
2. Руководство по работе в программной среде *RoboGuide V 6.40*, режим доступа: <http://belfingroup.com/o-belfingrupp/spravka/skachat/pdf-katalogi-funuc.html>. (дата обращения 24.02.2024г.).

### 3. ПРОГРАММИРОВАНИЕ СВАРОЧНОГО РОБОТА KUKA

Цель – приобретение навыков программирования промышленного сварочного робота *KUKA*.

С этой целью необходимо изучить методику программирования робота *KUKA KR 8 R1620*, оснащенного сварочным оборудованием для сварки в среде защитного газа *MIG-MAG*, и выполнить программирование движения сварочной горелки по сложной пространственной траектории.

#### 3.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ

Для того, чтобы приступить к программированию робота *KUKA*, необходимо изучить и получить начальные навыки работы с переносным пультом управления *smartPAD*. Кроме этого необходимо выполнить калибровку декартовых систем координат и нагрузки [1].

Программирование робота выполняется комбинированным методом, включающим элементы метода обучения и эвристического метода. Метод обучения предполагает перемещение инструмента робота последовательно в те точки пространства, где он должен побывать с последующей записью координат этих точек в программу. При этом используются клавиши ручного управления пульта *smartPAD*. Эвристический метод предполагает запись в программу типа и параметров движения, команды на внешние устройства, логические команды и т.п. с использованием специального языка программирования *KRL*.

После включения контроллера робота и загрузки системы на экране пульта *smartPAD* появляется навигатор, аналогичный интерфейсу операционной системы *Windows* (рис. 3.1). В левой половине экрана изображено дерево папок системы, а справа – перечень программных модулей в папке Program. На периферии экрана расположены сенсорные функциональные клавиши и иконки. Они активно используются при программировании робота. Их описание представлено в табл. 3.1.

Программирование начинается с определения статуса пользователя. Для этого касаемся иконки 13 и в появившемся списке пользователей выбираем статус **Эксперт**. Для закрепления этого

статуса вводим пароль (*KUKA*). Эксперт наделён правом создавать новые программы. После этой процедуры каждый из программных модулей в правой части экрана разделился на две про-

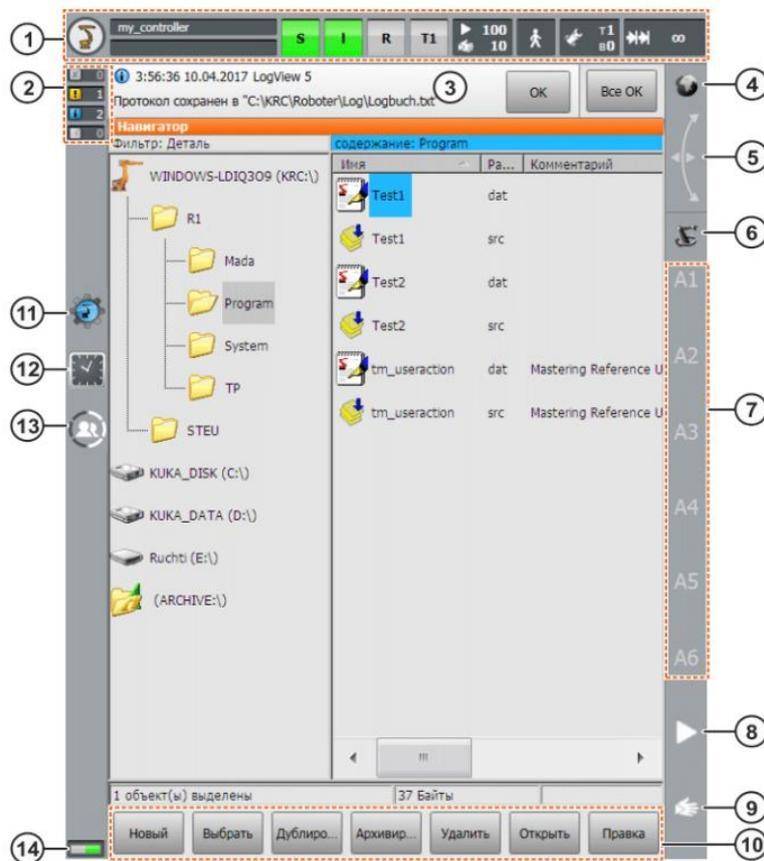


Рис. 3.1 Пользовательский интерфейс *KUKA smartNMI*

граммы с расширением **dat** и **src**. Программа с расширением **dat** формируется автоматически. В ней накапливается вся цифровая информация о координатах точек и постоянных данных. В программе с расширением **src** формируется программный код.

Таблица 3.1

#### Функциональные клавиши и иконки

Поз.	Описание
1	Линейка состояний (см. стр. 29)
2	Счётчик сообщений. Показывает количество сообщений всех типов.
3	Окно сообщений. По умолчанию показывается только последнее сообщение. Нажатием <b>ОК</b> можно квитировать квитируемое сообщение. Нажатием <b>Все ОК</b> квитируются все квитируемые сообщения.
4	Индикация состояния <b>Пространственная мышь</b> . Показывает актуальную систему координат для перемещения вручную с помощью пространственной мыши. При касании этой индикации можно выбрать другую систему координат.

5	Индикация <b>Ориентация пространственной мыши</b> . Касание этой индикации открывает окно, отображающее текущую ориентацию пространственной мыши, которую можно изменить.
6	Индикация состояния <b>Клавиши перемещения</b> . Показывает текущую систему координат для перемещения с помощью клавиш. При касании этой индикации можно выбрать другую систему координат.
7	Надписи на клавишах перемещения. Если выбрано осевое перемещение, здесь показываются номера осей ( <b>A1</b> , <b>A2</b> и т.д.). Если выбрано перемещение в прямоугольных координатах, здесь показываются направления системы координат ( <b>X</b> , <b>Y</b> , <b>Z</b> , <b>A</b> , <b>B</b> , <b>C</b> ).
8	Коррекция скорости выполнения программы в процентах от максимальной.
9	Коррекция скорости ручного перемещения в процентах от максимальной.
10	Панель функциональных клавиш. Изменяются динамично в зависимости от активного в настоящий момент окна в интерфейсе <i>smartHMI</i> . При касании клавиши <b>Правка</b> можно вызвать различные команды, относящиеся к навигатору.
11	Символ <b>WorkVisual</b> При его касании выполняется переход к окну управления проектами.
12	Часы. Показывают системное время и текущую дату.
13	Символ <b>группа Пользователя</b> . Количество белых сегментов в круге показывает, какая группа пользователей выбрана в данный момент.
14	Индикация активности. Если индикация мигает зелёным светом, значит интерфейс <i>smartHMI</i> активен.

Программу с расширением **dat** открывать не рекомендуется. Все команды программного кода записываются в программу с расширением **src**.

Перед тем как приступить к разработке программы необходимо изучить линейку состояний робота (поз. 1 на рис. 3.1) (рис. 3.2).

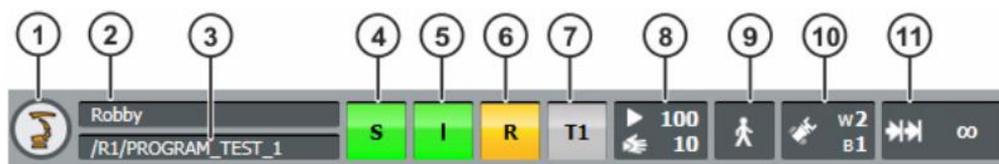


Рис. 3.2 Линейка состояний *KUKA smartHMI*

При касании большинства индикаций открывается окно, в котором можно изменить данный параметр состояния. Описание индикаций представлено в табл. 3.2.

Таблица 3.2

**Параметры состояния робота**

Поз.	Описание
1	Кнопка вызова главного меню. При выполнении данной работы вызов главного меню не потребуется.
2	Имя системы управления роботом
3	Имя программы
4	Индикация состояния <b>Импретатор заданий</b> .
5	Индикация состояния <b>Приводы</b> . При касании данной индикации открывается окно, в котором можно включить или выключить приводы.
6	Индикация состояния <b>Импретатор робота</b> . Здесь можно выполнить сброс программы на её начало или отменить выбор программы.
7	Актуальный режим работы. Эта индикация устанавливается не в линейке состояний. В данной работе используется тестовый режим <b>T1</b> , при котором в целях безопасности ограничена максимальная скорость (250 мм/сек).
8	Индикация состояния <b>Коррекция скорости</b> при выполнении программы и при ручном перемещении.
9	Индикация состояния <b>Вид выполнения программы</b> . Возможно четыре вида выполнения программы: пошаговый, непрерывный, выполнение одного шага и обратное выполнение.
10	Индикация состояния <b>Текущий инструмент и база</b> . Показывает номера выбранных систем координат инструмента и базы.
11	Индикация состояния <b>Инкрементальное перемещение вручную</b> . При касании индикации возможен выбор инкремента в миллиметрах или в градусах.

**Создание нового программного модуля.** Программный модуль создаётся в папке **Program**. Для этого необходимо открыть эту папку и коснуться сенсорной клавиши **Новый** (рис. 3.1) и на появившейся клавиатуре набрать имя нового модуля с подтверждением нажатием клавиши **ОК**. В списке навигатора появятся две программы с одинаковым новым именем, но с разными расширениями **dat** и **src**.

Далее, активировав курсором программу с расширением **src**, коснуться сенсорной клавиши **Открыть**. На экране появится шаблон будущей программы:

**DEF MAINPROGRAM** (имя программы)

**INI** (инициализация переменных)

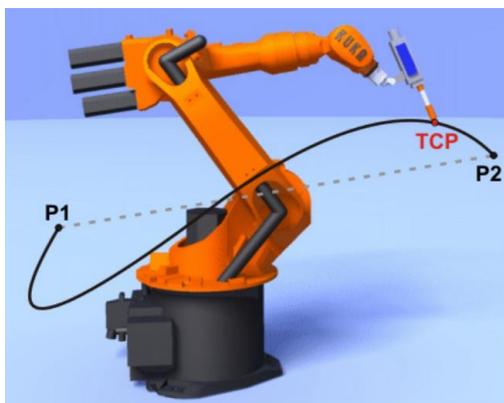
**SPTP HOME Vel= 100% DEFAULT** (выход в домашнюю точку)

**SPTP HOME Vel= 100% DEFAULT** (возвращение в домаш. точку)

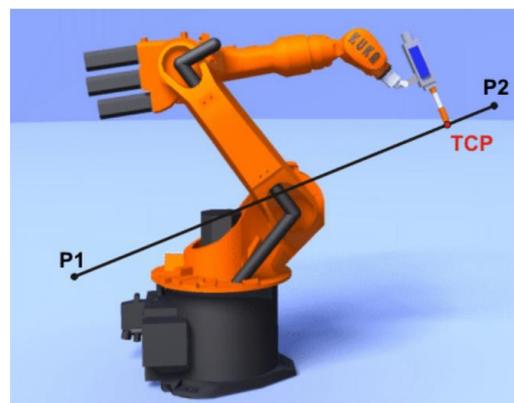
**END**

Вторая строка шаблона может дополняться описанием типа переменных, если в этом есть необходимость. Основной текст программы должен располагаться между одинаковыми командами **SPTP HOME**. Команда **END** завершает программу. Удалять эту команду запрещено! Исходное домашнее (**HOME**) положение является единым во всех программах данного робота. Поэтому его изменять запрещено!

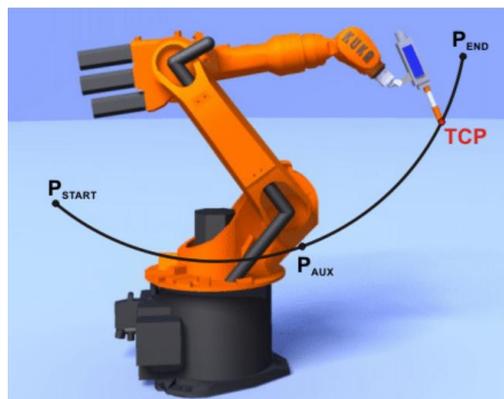
**Программирование движения.** Для программирования движения доступны три основных вида перемещения: **SPTP** (от точки к точке), **SLIN** (по прямой линии), **SCIRC** (по дуге окружности) (рис. 3.3) и дополнительное перемещение **SPLINE** (по кривой линии).



**SPTP**



**SLIN**



**SCIRC**

Рис. 3.3 Основные виды перемещения

Робот при движении по команде **SPTP** «переносит» точку **ТСР** по наиболее быстрой траектории. Наиболее быстрой траекторией является не прямая. Так как звенья робота выполняют вращательные движения, криволинейные траектории могут выполняться быстрее, чем прямые. Это необходимо учитывать и быть осторожным при программировании больших пространственных перемещений, чтобы на траектории перемещения не оказалось препятствия.

Движение по команде **SLIN** выполняется с определённой скоростью по отрезку прямой к целевой точке **P2**. Движение **SCIRC** выполняется с определённой скоростью вдоль круговой траектории к целевой точке. Круговая траектория задаётся начальной **P<sub>START</sub>**, вспомогательной **P<sub>AUX</sub>** и целевой точкой **P<sub>END</sub>**. Всегда начальная точка любого типа движения является целевой для предыдущей траектории движения.

Дополнительное движение **SPLINE** (рис. 3.4) программируется в виде блока (массива) в котором описываются все промежуточные точки, через которые должна пройти траектория.

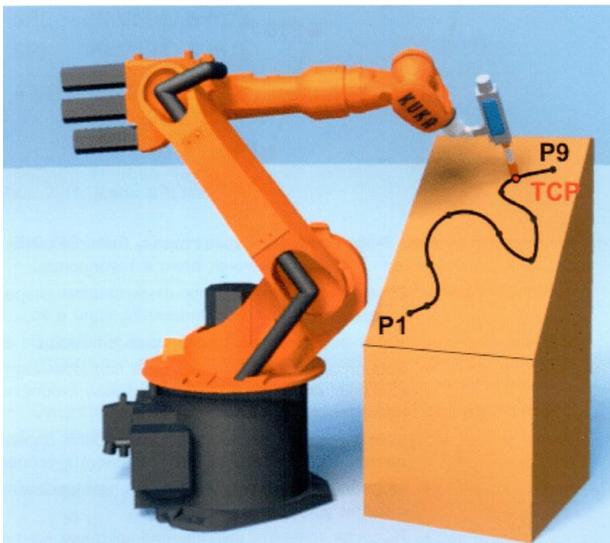


Рис. 3.4 Движение **SPLINE**

Структура блока **SPLINE** представлена ниже. Красным цветом отмечено имя блока **SPLINE**.

```

SPLINE S1 Vel=0.5 m/s Tool [1]
...
SPL P1 Vel=0.3 m/s
SPL P2 Vel=0.3 m/s
SPL P3 Vel=0.5 m/s
...
SPL P9 Vel=0.5 m/s
ENDSPLINE

```

**Программирование перемещений с помощью встроенных формуляров.** После того, как вызван шаблон программы, необходимо поместить курсор на первую строку исходной домашней точки **SPTP HOME**. Эта точка будет являться начальной для программируемого движения. Затем, используя клавиши ручного перемещения (поз. 7 на рис. 3.1) или пространственную

мышь, перевести точку **TCP** в целевую точку, если программируется движение **SPTP** или **SLIN**. Далее необходимо вызвать формуляр будущей команды нажатием клавиши **Команды** ((2) на рис. 3.5) и выбором в строке **Перемещение** необходимой команды (3).

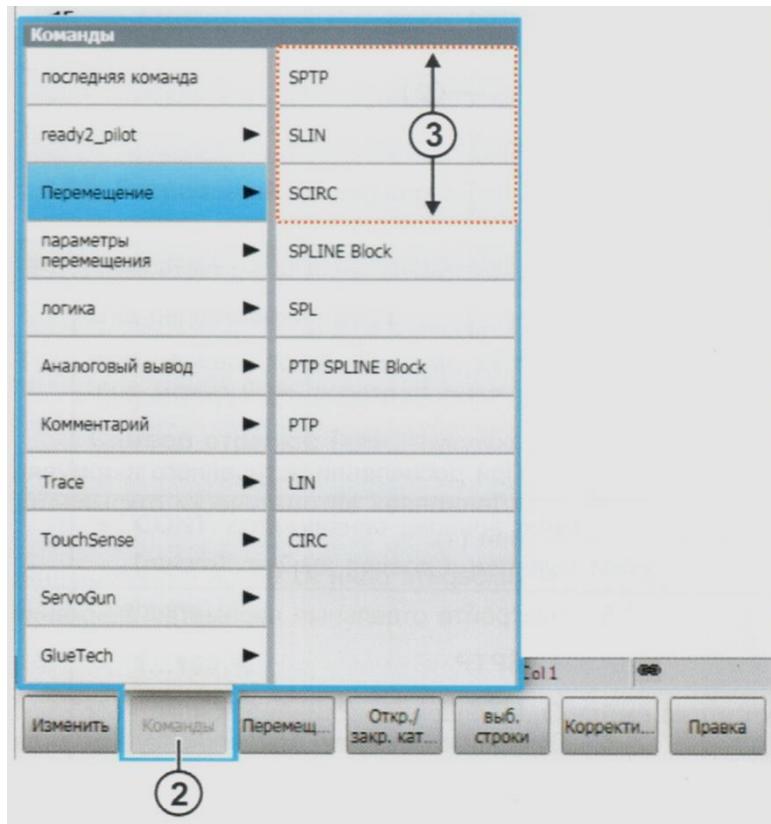


Рис. 3.5 Добавление встроенного формуляра

После этого появится частично заполненный встроенный формуляр команды, например, **SPTP** (рис. 3.6).

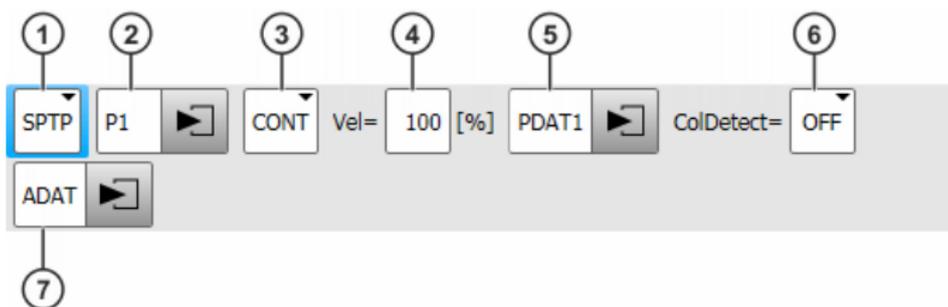


Рис. 3.6 Встроенный формуляр **SPTP**

Какие поля необходимо заполнить или скорректировать показано в табл. 3.3. На начальном этапе заполнения полей формуляра имеется возможность изменить вид перемещения (1) из всплывшего меню (2) (рис. 3.7, а).

### Параметры формуляра SPTP

Поз.	Описание
1	Вид перемещения <b>SPTP</b>
2	Имя и номер целевой точки (задаётся автоматически, может быть перезаписано). Касание символа стрелки позволяет выбрать откалиброванные системы координат (фреймы) инструмента и базы (рис. 3.7, б).
3	<b>CONT</b> сглаживание целевой точки [пусто] – точный подвод в целевую точку (рис. 3.7, в).
4	Скорость перемещения: для <b>SPTP</b> – 1...100%; для <b>SLIN</b> – 0.001...2 m/s.
5	Имя записи данных перемещения (задаётся автоматически, может быть перезаписано). Касание символа стрелки позволяет выбрать эти данные (рис. 3.7, з).
6	Распознавание столкновения для данного перемещения: <b>OFF</b> – функция распознавания столкновения отключена; <b>CDSets_Set [№]</b> – функция распознавания столкновения с заданным номером данных включена.
7	Имя записи данных с логическими параметрами.

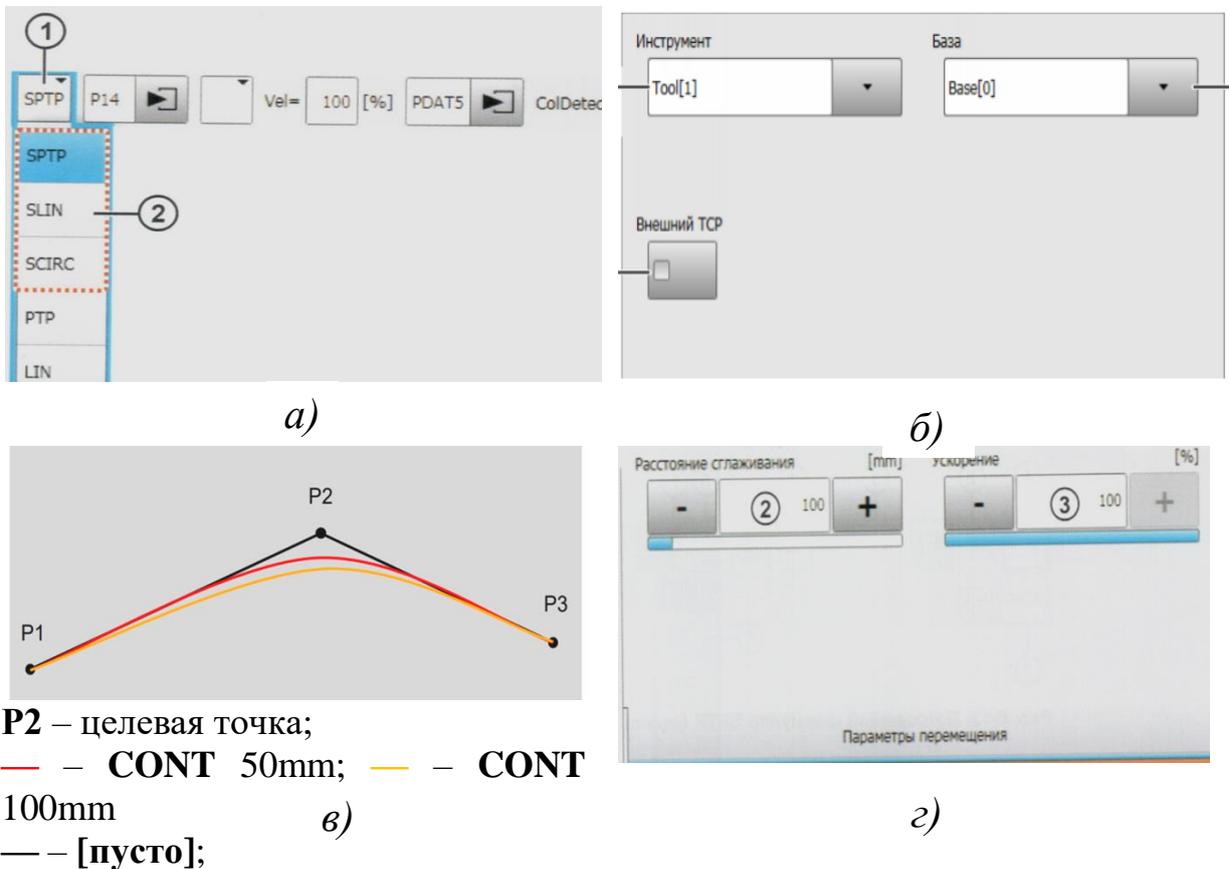


Рис. 3.7 Параметры формуляров SPTP и SLIN

После заполнения полей формуляра команда записывается нажатием клавиши **Команда ОК**.

**Формуляр SCIRC** отличается от формуляра **SPTP** или **SLIN** введением вспомогательной точки  $P_{AUX}$  (рис. 3.3) и угла, стягивающего дугу окружности (2) и (8) на рис. 3.8.

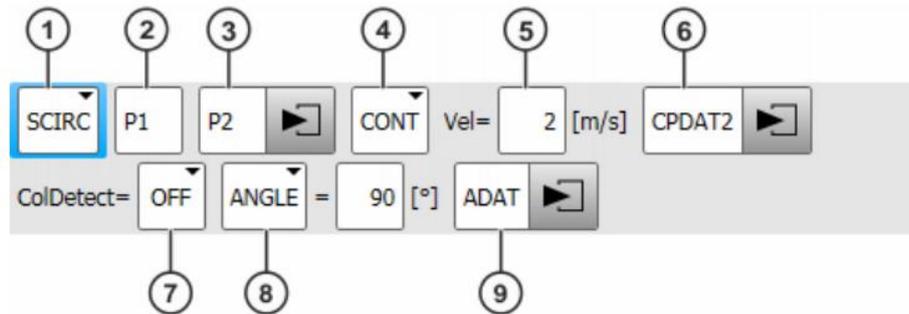


Рис. 3.8 Встроенный формуляр **SCIRC**

На начальном этапе заполнения полей формуляра имеется возможность изменить вид перемещения (1) из всплывшего меню (2) (рис. 3.7, а). Остальные поля описаны в табл. 3.4.

Таблица 3.4

Параметры формуляра **SCIRC**

Поз.	Описание
1	Вид перемещения <b>SCIRC</b>
2	Имя и номер вспомогательной точки (задаётся автоматически, может быть перезаписано).
3	Имя и номер целевой точки (задаётся автоматически, может быть перезаписано). Касание символа стрелки позволяет выбрать откалиброванные системы координат (фреймы) инструмента и базы (рис. 3.7, б).
4	<b>CONT</b> сглаживание целевой точки <b>[пусто]</b> – точный подвод в целевую точку (рис. 3.7, в).
5	Скорость перемещения: <b>0.001...2 m/s</b> .
6	Имя записи данных перемещения (задаётся автоматически, может быть перезаписано). Касание символа стрелки позволяет выбрать эти данные (рис. 3.7, г).
7	Распознавание столкновения для данного перемещения: <b>OFF</b> – функция распознавания столкновения отключена; <b>CDSets_Set [№]</b> – функция распознавания столкновения с заданным номером данных включена.
8	Угол, стягивающий дугу окружности: <b>- 9999° ... + 9999°</b> .
9	Имя записи данных с логическими параметрами.

Перед записью команды **SCIRC** необходимо в ручном режиме вывести точку **TCP** во вспомогательную точку и нажать клавишу **Вспомогательная точка**. То же самое выполнить с целевой точкой. И только после введения угла, стягивающего дугу окружности, записать команду нажатием клавиши **Команда ОК**.

Программирование движения по сплайн-траектории начинается с вызова встроенного формуляра **SPLINE Block** (Рис. 3.5). При этом точка **TCP** должна находиться в начале сплайн-траектории. Встроенный формуляр заголовка **SPLINE Block** представлен на рис. 3.9.

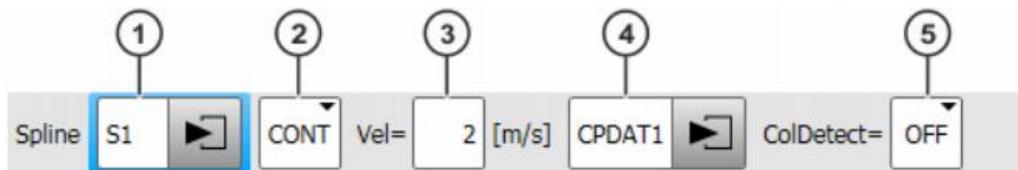


Рис. 3.9 Встроенный формуляр заголовка **SPLINE Block**

Все поля в заголовке блока настраиваются так же, как показано в табл. 3.3. Все промежуточные точки, входящие в сплайн-блок, записываются командой **SPL** (рис. 3.5) перед командой закрытия блока **ENDSPLINE**. Через некоторое время после закрытия сплайн-блока его содержимое (кроме заголовка) скрывается. Для того, чтобы его снова открыть, необходимо навести курсор на заголовки сплайн-блока (1) (рис. 3.10) и нажать клавишу **Откр./закр. кат...** (2).

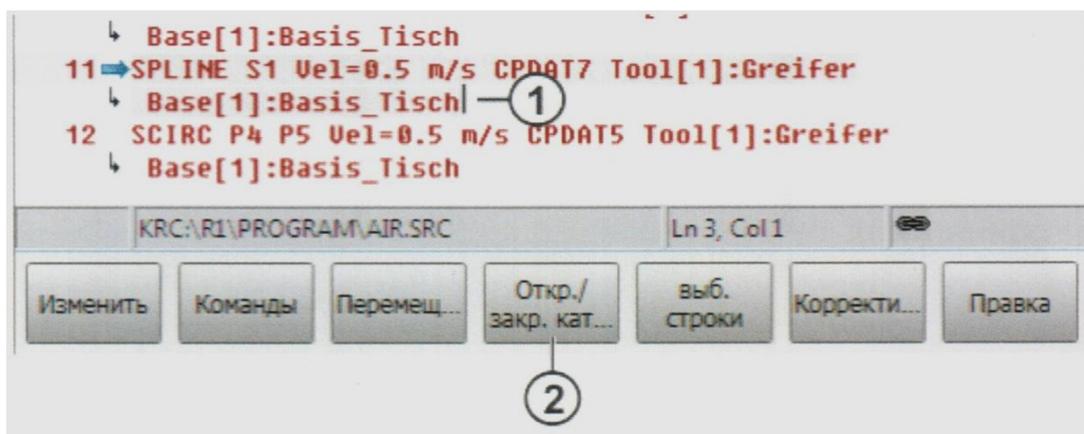


Рис. 3.10 Открытие/закрытие сплайн-блока

### 3.2. ЗАДАНИЕ

Разработать и отладить программу движения точки ТСП горелки сварочного робота *KR 8 R1620* по сложной траектории, содержащей все виды движения: от точки к точке **SPTP**, по отрезку прямой линии **SLIN**, по дуге окружности **SCIRC** и по сплайн-траектории **SPLINE**. Имя программы задать по фамилии обучающегося.

Продемонстрировать преподавателю в тестовом режиме Т1 работу робота по составленной программе.

### 3.3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Назовите методы программирования промышленных роботов.
2. Какой тип движения подходит для программирования выхода точки ТСП из домашнего положения в рабочую точку?
3. Какова конфигурация траектории точки ТСП при обработке команды **SPTP**?
4. Каковы преимущества использования параметра сглаживания?
5. Как назначить откалиброванные фреймы при программировании движения?
6. Какова структура **SPLINE Block**?

### 3.4. ЛИТЕРАТУРА

1. Калибровка сварочного робота КУКА : методические указания к практическим занятиям для обучающихся направления подготовки 15.04.04 "Автоматизация технологических процессов и производств", профиль "Роботы и робототехнические системы", всех форм обучения / составитель Н. П. Курышкин. – Кемерово : КузГТУ, 2022, – 1 файл (545 Кб), режим доступа : <http://library.kuzstu.ru/meto.php?n=10466>. Текст электронный.
2. *KUKA System Software 8.3 RUS* Инструкция по эксплуатации и программированию для конечного пользователя, режим доступа: [studylib.ru](http://studylib.ru) (дата обращения 24.09.2023г.).
3. *KUKA industrial Robot*, режим доступа: <http://www.kuka-industries.com/de/career/> (дата обращения 24.09.2023г.).

## 4. ПРОГРАММИРОВАНИЕ ПРОМЫШЛЕННОГО РОБОТА FANUC

Цель – приобретение навыков программирования промышленного робота *FANUC*.

С этой целью необходимо изучить методику программирования промышленного робота *FANUC ER-4iA*, оснащённого электрическим двухпальцевым захватом *SCHUNK* и входящего в состав учебной роботизированной ячейки. Используя эту методику, разработать и отладить программу загрузки импровизированного технологического оборудования по предложенной преподавателем пространственной траектории.

### 4.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ

Для того, чтобы приступить к программированию робота *FANUC*, необходимо изучить и получить начальные навыки работы с переносным пультом управления *iPendant*. Кроме этого необходимо выполнить настройку декартовых систем координат и нагрузки [1].

Программирование робота выполняется комбинированным методом, включающим элементы метода обучения и эвристического метода. Метод обучения предполагает перемещение инструмента робота последовательно в те точки пространства, где он должен побывать с последующей записью координат этих точек в программу. При этом используются клавиши ручного управления пульта *iPendant*. Эвристический метод предполагает запись в программу типа и параметров движения, команды на внешние устройства, логические команды и т.п. с использованием специального языка программирования *Karel*.

Создание новой программы состоит из трёх этапов. На первом этапе вводится имя программы и формируется поле для ввода команд. На втором этапе описываются системы координат и нагрузка, которые будут использоваться в данной программе. На третьем, заключительном этапе, вводятся команды с использованием метода обучения.

Для создания программы и начала программирования на первом этапе необходимо на переносном пульте управления *iPendant* нажать клавишу *SELECT*, после чего на экране отобразится пере-

чень уже записанных программ (рис. 4.1). В данном списке присутствует набор системных программ, защищённых от записи и редактирования. Они имеют некий комментарий в скобках справа.

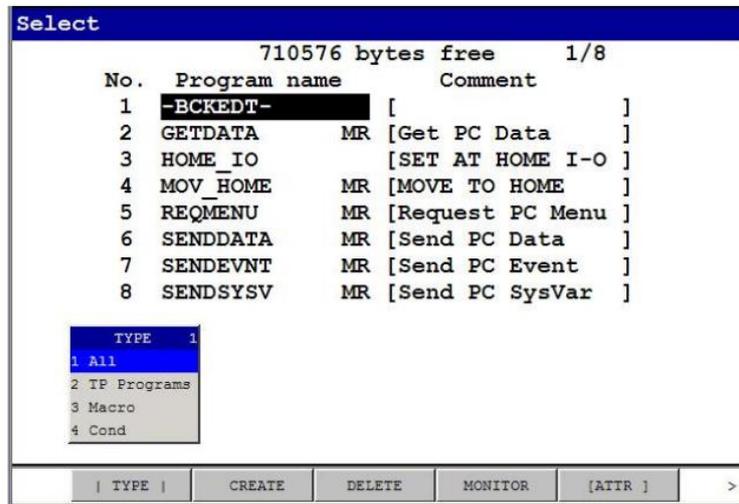


Рис. 4.1 Окно списка программ

В нижнем ряду экрана появились функциональные сенсорные клавиши. Они продублированы физическими клавишами  $F1$ , ...,  $F5$ . Если нажать сенсорную клавишу  $[TYPE]$  (или  $F1$ ), то можно сузить список отображаемых программ, выбрав из всплывающего меню следующее (рис. 4.1):

- 1 *All*: отображение всех программ;
- 2 *TP Programs*: отображение всех программ с расширением *TP*;
- 3 *Macro*: отображение макросов (миниподпрограмм);
- 4 *Cond*: отображение всех программ формата *Cond*.

Чтобы создать новую программу, необходимо нажать либо физическую клавишу  $F2$ , либо сенсорную – *CREATE*. После этого откроется окно, где необходимо задать имя программы (рис. 4.2).

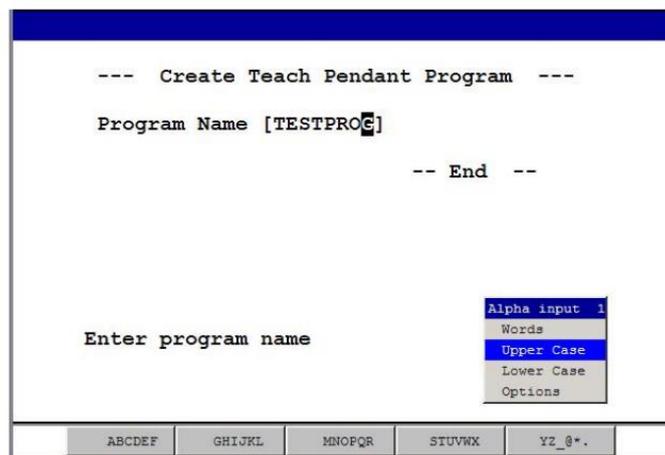


Рис. 4.2 Окно ввода имени программы

Имя программы записывается на клавиатуре, которая «всплывёт», если нажать последовательно функциональную клавишу *F5* и далее в появившемся меню – *Option/Keybrd*. Далее нажать клавишу *ENTER*. После этого откроется пустой экран для ввода текста программы.

Перед записью текста программы необходимо зарезервировать число пустых строк для команд программы. Для этого нажать последовательно *F3 [EDIT]* → *F5 [EDCMD]*, в появившемся меню – *Insert*. Далее ввести число пустых строк и нажать *ENTER*. Число пустых строк должно быть больше, чем примерное число команд новой программы. В процессе формирования программы число пустых строк можно добавлять. На этом заканчивается первый этап формирования новой программы.

На втором этапе необходимо задать номера тех систем координат инструмента (*UTOOL*), детали (*UFRAME*) и нагрузки (*PAYLOAD*), которые будут использоваться в данной программе. Для этого на сенсорном экране касаемся клавиши *NEW INSTRUCTION*. В появившемся меню выбираем строку *OFFSET FRAMES*, а в подменю – *UTOOL NUM = ...*. Далее вводим как *константу* номер системы координат инструмента. Аналогично вводится номер пользовательской системы координат (СК детали): *NEW INSTRUCTION* → *OFFSET FRAMES* → *UFRAME NUM = ...*. Далее вводим как *константу* номер системы координат детали. Номер нагрузки вводим нажатием функциональной клавиши *F1 [INST]*, а в меню выбираем *PAYLOAD [...]* и вводим номер нагрузки.

В процессе выполнения второго этапа на экране пульта появятся начальные строки будущей программы:

```

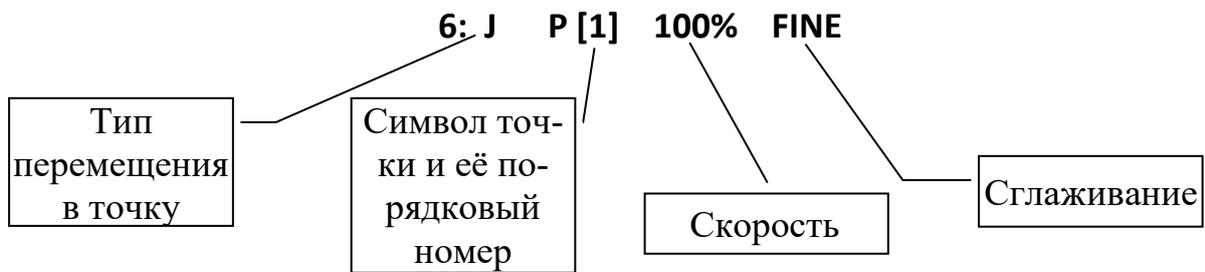
1:   !Init
2:   UTOOL_NUM = 1
3:   UFRAME_NUM = 1
4:   PAYLOAD[1]
5:
6:

```

Второй этап заканчивается обязательным запуском программы, чтобы указанные системы координат вступили в силу. Для этого необходимо снять клавишей *RESET* запрет на запуск

(красная лампочка *FAULT* на пульте должна погаснуть). Далее одновременно нажать клавиши *SHIFT* и *FWD*. Клавиша подтверждения с обратной стороны пульта должна быть нажата.

Третий этап разработки программы является основным. Здесь программируется движение робота. Стандартный шаблон команды перемещения состоит из четырёх основных элементов:



**Тип перемещения в точку.** Система предусматривает три типа перемещения:

**J** – перемещение в точку без контроля траектории с минимизацией времени отработки команды;

**L** – перемещение строго по отрезку прямой линии;

**C** – перемещение по дуге окружности

На рис. 4.3 представлены эти типы перемещения и команды на их отработку.

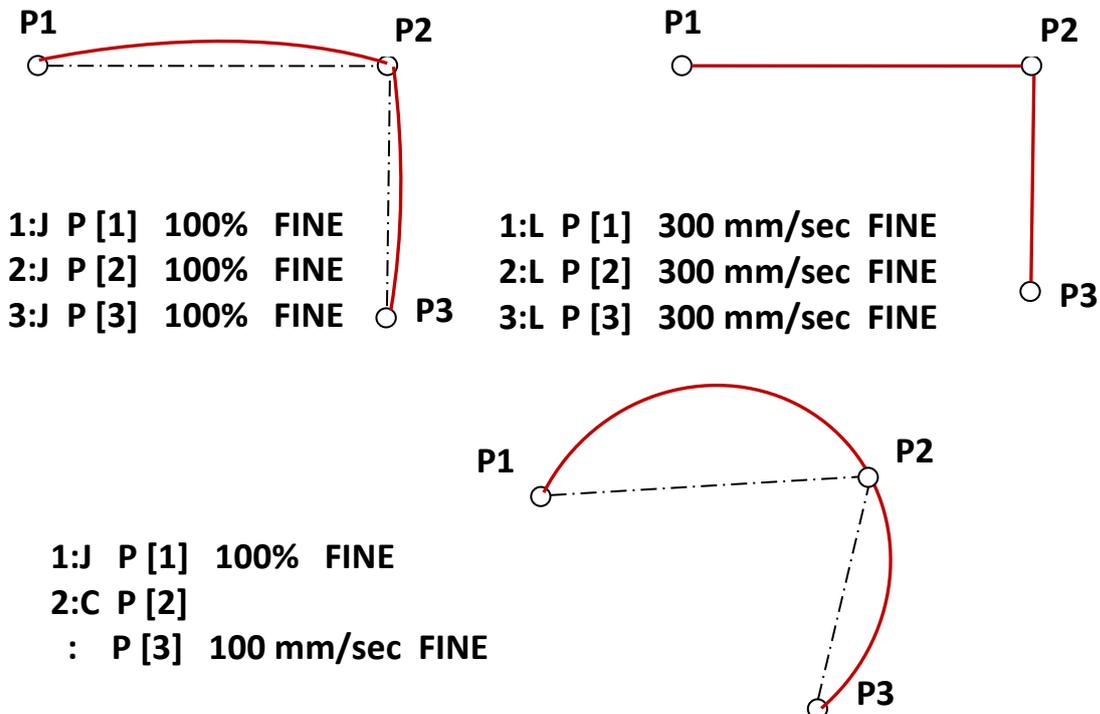


Рис. 4.3 Типы перемещения и их программирование

**Символ точки и её порядковый номер.** Чтобы запрограммировать какое-либо движение к точке, необходимо в ручном режиме перевести центральную точку *TCP* инструмента в требуемое положение. Затем нажать функциональную клавишу *F1[POINT]* (если клавише *F1* соответствует функция *[INST]*, то нажать *F5 >*). Далее выбрать из меню типа движения (рис. 3.4) нужный тип движения и нажать *ENTER*.

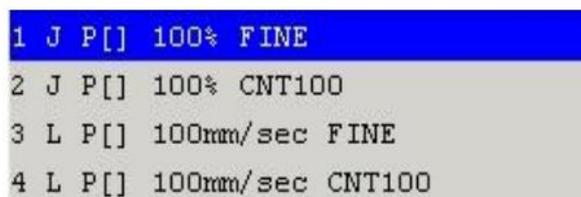


Рис. 4.4 Меню движения

Во всплывающем меню выбрать *C*. Таким же образом можно менять остальные элементы.

Точка с одним и тем же номером может повторяться в тексте программы. При этом сохраняются координаты её положения, а скорость, тип перемещения и сглаживание могут меняться. Если курсором навести на номер точки и нажать *ENTER*, появится возможность ввода имени точки.

**Скорость.** Скорость типа движения *J* задаётся в процентах от максимальной. Скорость линейного и кругового движения задаётся в мм/с. Эту размерность можно менять (*F4[CHOICE]*).

**Сглаживание.** Параметр *FINE* означает точный выход в запрограммированную точку с кратковременной в ней остановкой.

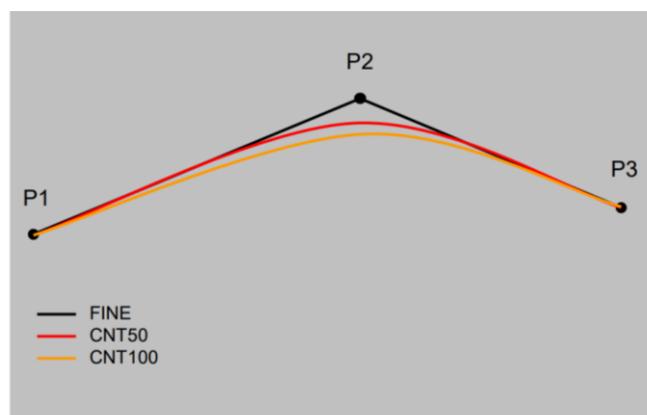


Рис. 4.5 Сглаживание траектории

Параметр *CNT* сглаживает траекторию без кратковременной остановки (рис. 4.5). Чем выше процент сглаживания, тем дальше от точки выполняется сглаживание траектории.

Следует соблюдать предельное внимание при использовании элемента сглаживания, так как он напрямую влияет на обработку траектории роботом. Любое изменение коэффи-

циента *CNT* или скорости перемещения влечёт за собой изменение траектории!

Окончательная запись команды перемещения выполняется одновременным нажатием клавиш *SHIFT* и *F5[RECORD]*. Если горит красная лампочка *FAULT* на пульте, то надо снять запрет нажатием клавиши *RESET*.

**Программирование захвата.** Электрическое захватное устройство робота взаимодействует с его контроллером через восемь бинарных входов и восемь бинарных выходов. Коннектор этих входов/выходов (*Input/Output*) смонтирован на подвижном звене манипулятора и соединён гибким кабелем с электромагнитами захвата. Для программирования захвата используются выходные сигналы (*Output*), подаваемые на контакты **7** и **8** коннектора. Эти сигналы работают в противофазе, т.е. если на контакт **7** подаётся сигнал *ON*, то на контакте **8** должен быть сигнал *OFF* и наоборот. В первом случае губки схвата откроются, а во втором – закроются.

При программировании захвата необходимо учесть, что на открытие и закрытие его губок необходимо некоторое время. Поэтому вначале и в конце этих команд необходимо предусмотреть выдержку времени. Именно с этой команды и начинается программирование схвата. Для этого нажать клавишу *F1[INST]* и выбрать из списка инструкций *WAIT* (рис. 4.6).



Рис. 4.6 Меню инструкций

Далее из подменю выбрать **WAIT ... sec** и нажать *ENTER*. Используя числовую клавиатуру, ввести время в секундах и *ENTER*.

Для программирования сжатия губок захвата нажимаем  $F1[INST] \rightarrow I/O \rightarrow RO[...]=...$  (рис. 4.7). Присваиваем  $RO$  номер 7, а значение выбираем  $ON$ .

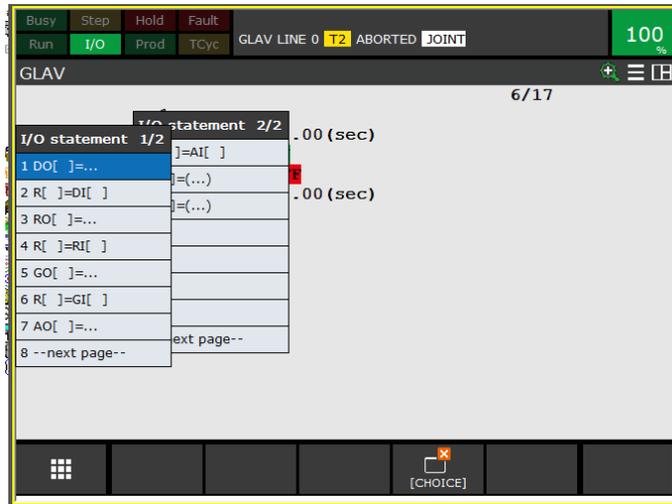


Рис. 4.7 Программирование сжатия губок

Подобным образом программируем выход 8. Только ему присваиваем значение  $OFF$ . Завершаем программирование захвата командой  $WAIT$  (рис. 4.8).



Рис. 4.8 Результат программирования

Для программирования разжатия губок захвата надо поменять значения  $RO[7]$  и  $RO[8]$  на инверсные.

На рис. 4.9 представлен фрагмент программы с именем  $GLAV$ . Программа заканчивается командой  $[End]$  Она завершает программу автоматически. Удалять эту команду запрещено! Во фрагменте первые три команды являются обязательными. Далее в четвертой команде программируется начальная (домашняя) точка  $P[1]$ .



Рис. 4.9 Фрагмент программы

С этой точки начинается движение. Она же должна быть и в конце программы. Символ @ в команде означает, что робот находится в данной точке.

## 4.2. ЗАДАНИЕ

Учебная ячейка оснащена промышленным роботом *FANUC ER-4iA* и вспомогательным оборудованием. В состав оборудования входит магазин деталей цилиндрической формы, находящихся в ячейках. Рядом с этим магазином закреплён другой магазин с пустыми ячейками.

Необходимо составить, отладить и продемонстрировать программу работы робота с одной деталью. Робот должен выйти из домашнего положения, взять указанную преподавателем деталь, поместить её в ячейку пустого магазина и вернуться в домашнюю точку. Далее через несколько секунд вернуть деталь в ячейку первого магазина и вернуться в домашнюю точку.

## 4.3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что необходимо выполнить перед программированием робота *FANUC*?
2. Чем отличается команда с типом движения **J** от команды **L**?
3. Как запрограммировать движение по дуге окружности?
4. Раскрыть последовательность программирования закрытия губок захвата.

#### 4.4. ЛИТЕРАТУРА

1. Настройка систем координат робота *FANUC* : методические указания к практическим занятиям для обучающихся направления подготовки 15.04.04 "Автоматизация технологических процессов и производств", профиль "Роботы и робототехнические системы", всех форм обучения ; составители: Н. П. Курьшкин, Н. В. Моисеев, Е. В. Бурков. – Кемерово. – КузГТУ, 2022.– 1 файл (750 Кб). – <http://library.kuzstu.ru/meto.php?n=10467> – Текст: электронный.
2. Официальный сайт компании *FANUC*, режим доступа [www.fanuc.eu/ru/ru](http://www.fanuc.eu/ru/ru) . (дата обращения 24.08.2023г.).
3. Руководство по работе в программной среде *RoboGuide V 6.40*, режим доступа: <http://belfingroup.com/o-belfingrupp/spravka/skachat/pdf-katalogi-funuc.html>. (дата обращения 24.08.2023г.).

## 5. СОЗДАНИЕ РОБОТИЗИРОВАННОЙ ЯЧЕЙКИ В ПРОГРАММНОЙ СРЕДЕ ROBOGUIDE

Цель – приобретение навыков по разработке в программной среде *ROBOGUIDE V9* виртуальной роботизированной ячейки.

С этой целью, используя программный комплекс *ROBOGUIDE V9* и приведённую ниже методику, по индивидуальному варианту создаётся роботизированная ячейка, включающая два промышленных робота *FANUC*, конвейер, ограждение и контроллер.

Работа выполняется на компьютере с загруженным пакетом *ROBOGUIDE V9* и рассчитана на два часа.

### 5.1. КРАТКИЙ ОБЗОР ПАКЕТА ROBOGUIDE V9

Для разработки и генерации управляющих программ функционирования промышленных роботов (ПР) в составе роботизированных технологических комплексов (РТК) и роботизированных ячеек фирма *FANUC* (Япония) вместе с роботами поставляет заказчику мощный, многофункциональный программный комплекс *ROBOGUIDE V9*. Этот пакет позволяет в полной мере использовать возможности 3D-моделирования. Он имеет развитую библиотеку, включающую весь типоразмерный ряд ПР и технологического оборудования, поставляемого фирмой *FANUC*. Кроме этого библиотека содержит: захватные устройства, технологический инструмент, рольганги, конвейеры, ограждения, средства безопасности и т.п. Если в библиотеке отсутствует какой-либо объект, то разработанную его *CAD*-модель можно импортировать в программу в формате *IGES*.

Пакет *ROBOGUIDE V9* имеет интуитивно понятный интерфейс и позволяет после разработки 3D-модели РТК запрограммировать в офлайн-режиме его работу, а симуляция этой работы на экране монитора позволяет быстро и эффективно выполнять отладку программы.

После разработки и отладки *Simulation*-программы работы РТК пакет *ROBOGUIDE V9* позволяет преобразовать её в исполняемую программу (*TP*-программу) в машинных кодах. Эту *TP*-программу посредством *flash*-памяти можно перенести в контроллер реального робота.

## 5.2. СОЗДАНИЕ РОБОТИЗИРОВАННОЙ ЯЧЕЙКИ

Самый эффективный способ научиться работать в какой-либо программной среде – это решить в этой среде реальную задачу с использованием пошагового путеводителя. Этот способ положен в основу методики усвоения пакета *ROBOGUIDE V9*.

Предлагается согласно выданному преподавателем варианту задания (см. табл. 5.1 на стр. 58), разработать виртуальную роботизированную ячейку, следуя приведённой ниже пошаговой инструкции.

### 5.2.1. Создание нового проекта

1. Запускаем программный комплекс *ROBOGUIDE V9*, дважды кликнув на значок с надписью *ROBOGUIDE* на рабочем столе.

Изучаем интерфейс программы, перемещая курсор по надписям всплывающих подменю основного меню программы.

2. Создаем свой (новый) проект (*New cell*). Для этого кликнем курсором мыши на значке главного меню (рис. 5.1).

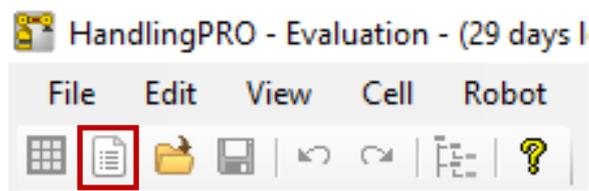


Рис. 5.1 Создание нового проекта

3. В открывшемся окне (рис. 5.2) введём имя проекта и нажмём *Next*>.

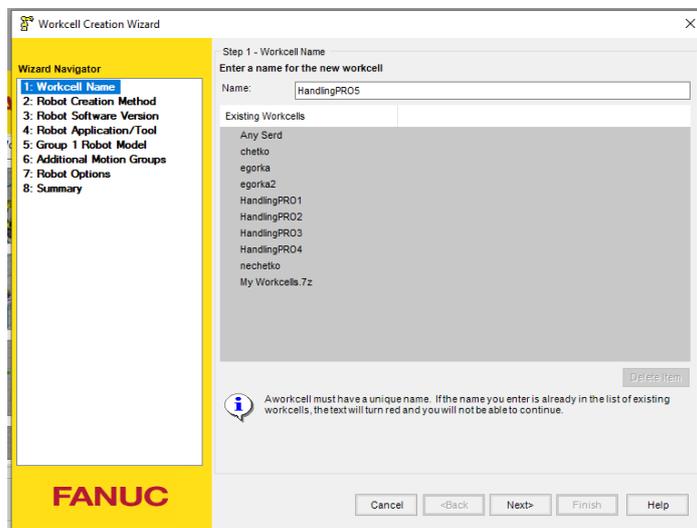


Рис. 5.2 Ввод имени проекта

Замечаем, что слева на жёлтом фоне показана нумерованная последовательность создания Вашего проекта с отметкой текущего активного шага.

4. В следующем окне (рис. 5.3) указываем способ создания нового робота – *Robot Selection Method*. Выбираем один из четырёх вариантов:

a) *Create a new robot with the default HandlingPro config* – создать нового робота с настройками по умолчанию;

b) *Create a new robot with the last used HandlingPro config* – создать нового робота с последними примененными настройками;

c) *Create a robot from a file backup* – создать робота из существующей резервной копии (*All of above*);

d) *Create an exact copy of an existing robot* – создать точную копию существующего робота, подключенного к сети.

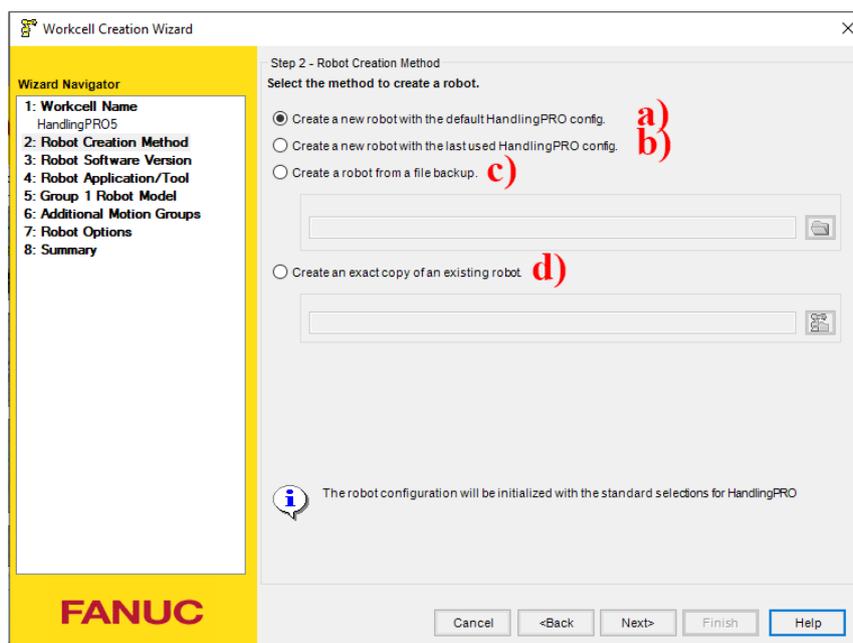


Рис. 5.3 Выбор робота

Выбираем – «создать нового робота с настройками по умолчанию» и нажимаем кнопку *Next>*.

5. На следующем, третьем шаге выбираем версию программного обеспечения контроллера (по умолчанию – последняя выпущенная версия). Нажимаем кнопку *Next>*.

6. На четвёртом шаге выбираем тип программного обеспечения (рис. 5.4). По умолчанию – *HandlingTool*. В случае сварочного применения выбираем – *ArcTool*. Далее нажимаем кнопку *Next>*.

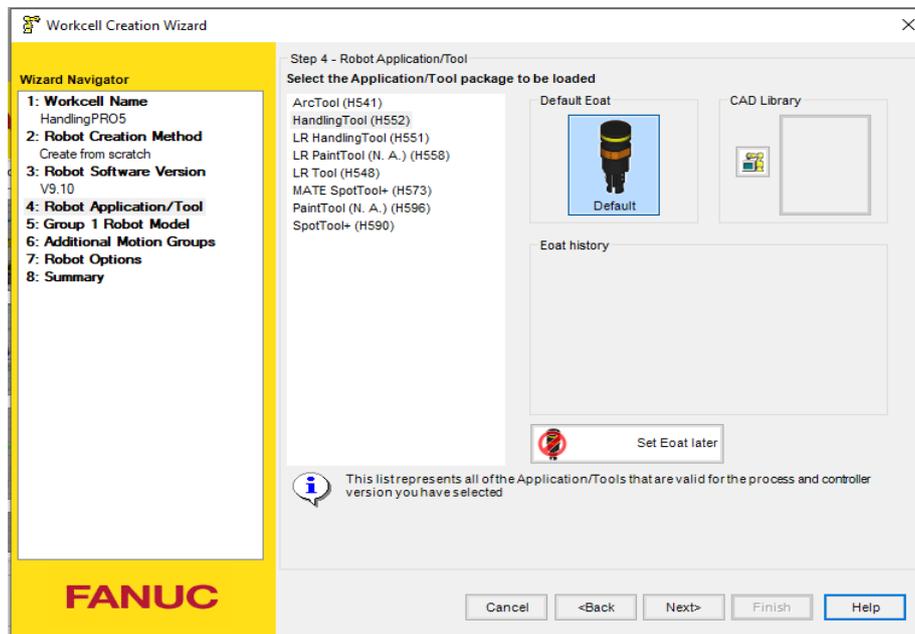


Рис. 5.4 Выбор типа программного обеспечения

7. На пятом шаге из типоразмерного ряда согласно варианту задания выбираем модель робота (рис. 5.5) и нажимаем кнопку *Next>*.

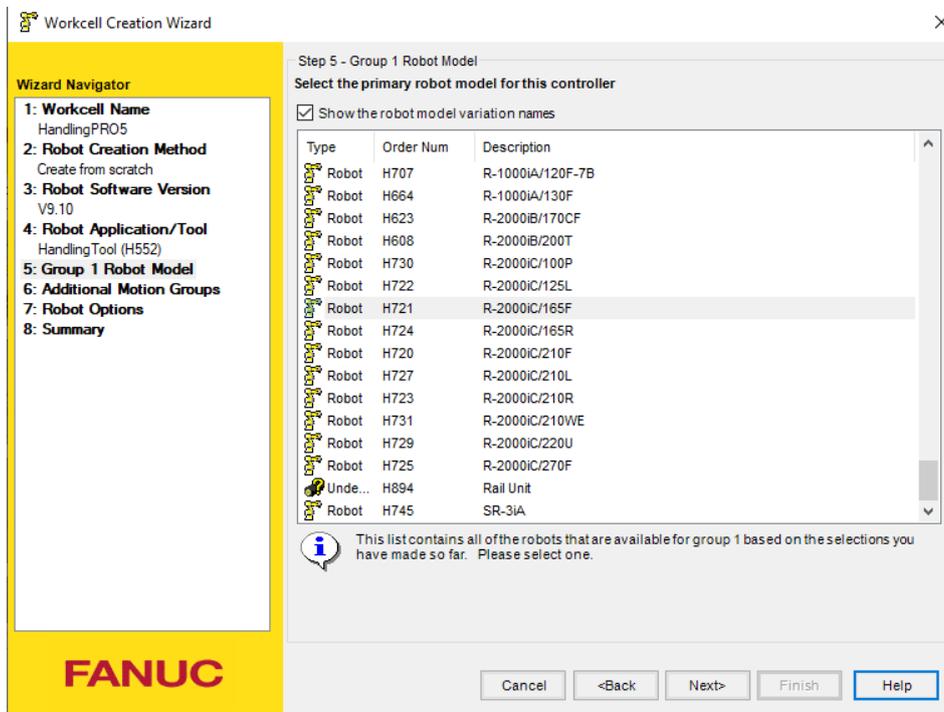


Рис. 5.5 Выбор модели робота

8. При необходимости мы можем добавить дополнительные группы движения “*Additional Motion Groups*” в виде позиционирующих устройств, сервоприводов или роботов. Это действие выполняется на шестом шаге. На этом шаге в проект добавляется ещё один робот согласно варианту задания. В нашем примере этого не требуется, поэтому нажимаем кнопку *Next*>.

9. Седьмой шаг создания проекта позволяет выбрать дополнительные опции программного обеспечения. Нам это не требуется, поэтому нажимаем кнопки *Next*> и *Finish* в следующем окне *Summary*.

После запуска виртуального контроллера у Вас попросят выбрать из двух вариантов присоединительного фланца робота. В нашем случае выбираем вариант *Standart* (клавиша *1*). После этого наш новый проект будет создан (рис. 5.6).

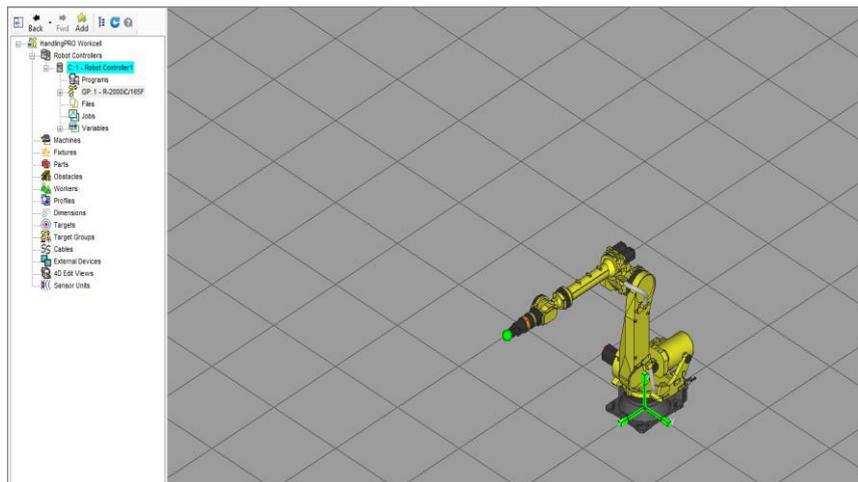


Рис. 5.6 Результат создания проекта

Созданный проект включает только робота. На следующих этапах этот робот оснащается инструментом, вспомогательным оборудованием, ограждением и т.п.

### 5.2.2. Добавление инструмента

1. Важным информационным элементом и путеводителем по проекту является дерево обозревателя проекта (*Cell Browser*). Необходимо, чтобы окно этого дерева было постоянно открыто в левой части экрана. Для его открытия нажмём на указанный значок (рис. 5.7). На этом этапе необходимо внимательно изучить дерево проекта, передвигая по нему курсором мыши.

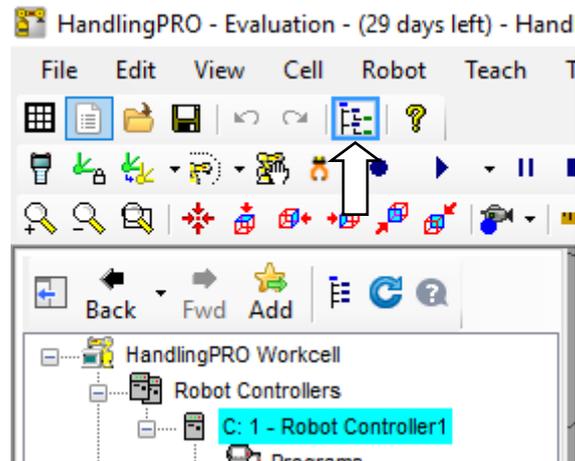


Рис. 5.7 Открытие дерева обозревателя

2. На втором шаге добавления инструмента откроем его свойства, как указано на рис. 5.8.

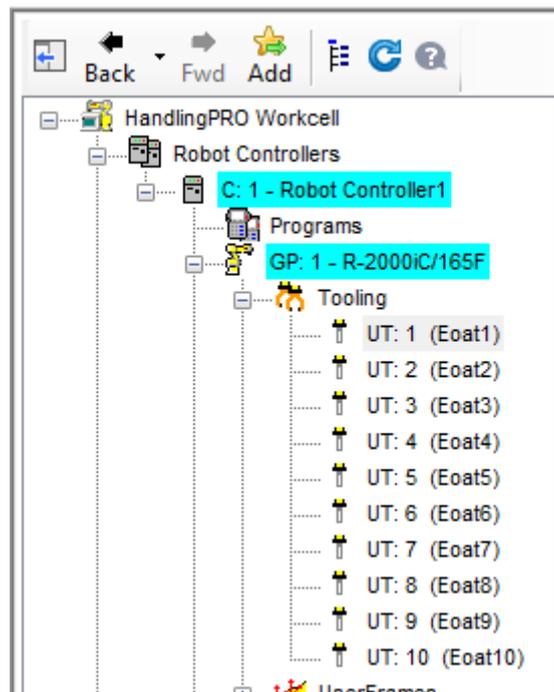


Рис. 5.8 Добавление инструмента

Двойным щелчком мыши на строке *UT:1 (EOAT1)* открываем (рис. 5.9) свойства данного инструмента (*EOAT = End of Arm Tool*). В строке *Name* присвоим инструменту имя латинскими буквами. По умолчанию – *Eoat1*.

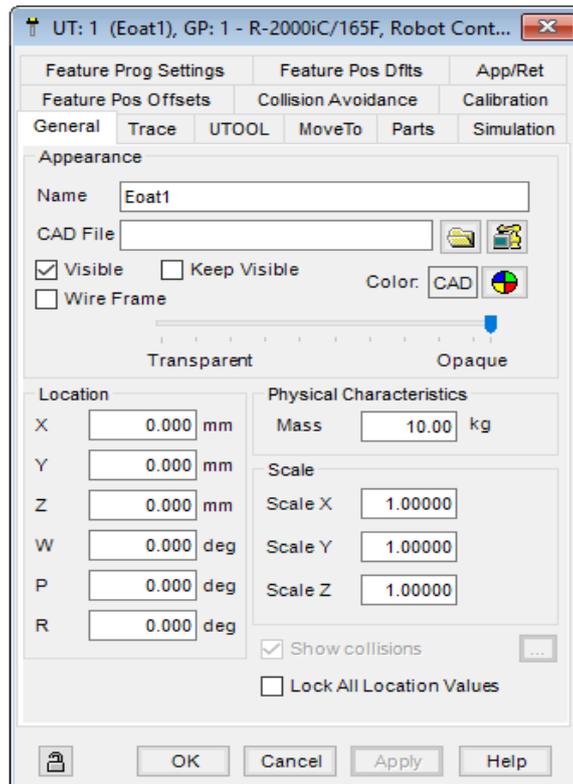
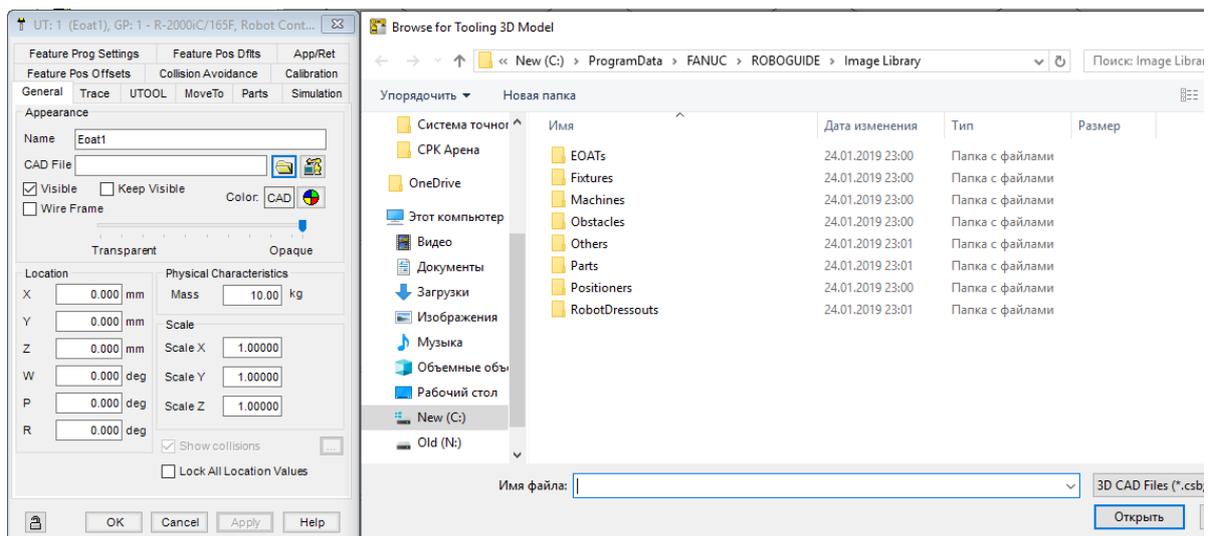


Рис. 5.9 Имя инструмента

Далее импортируем *CAD*-модель инструмента из библиотеки (*Image Library*) (рис. 5.10). Для этого откроем папку *EOAT's*, затем *Grippers* и выбираем файл *Vacuum01.IGS* – Вакуумный захват. Укажем его массу (kg).

Рис. 5.10 Импорт *CAD*-модели инструмента

После выбора *CAD*-модели инструмента станет активной кнопка *Apply*. Нажмём её. Модель инструмента добавлена к роботу (рис.5.11).

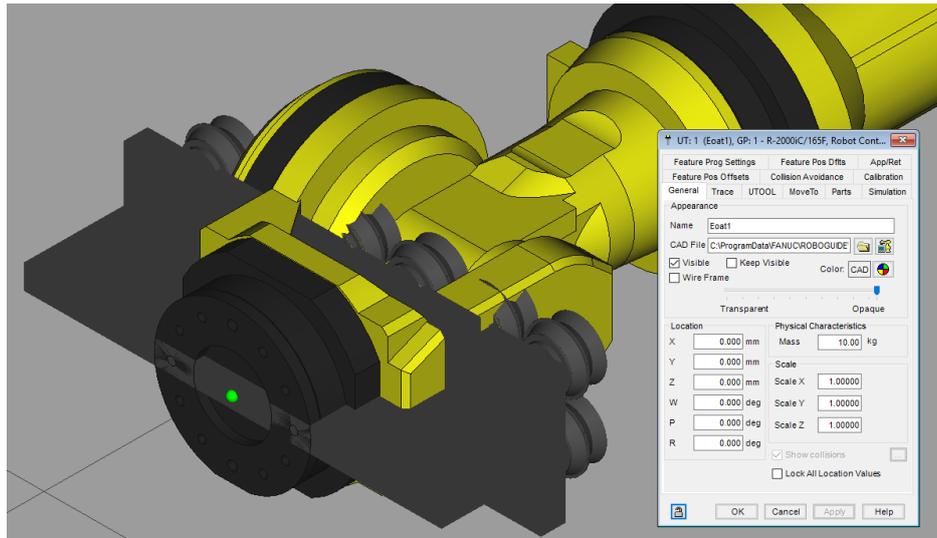


Рис. 5.11 Добавленный вакуумный инструмент

Однако замечаем, что присоски вакуумного захвата обращены в тело манипулятора. Для корректной установки инструмента необходимо в программе 3D моделирования, в которой создавалась модель, привязать объект к началу системы координат. Для этого в окне свойств инструмента (рис. 5.12) введём значение угла разворота инструмента вокруг оси  $x$  (ось  $w$ ), равное  $180^\circ$  (180.000 deg).

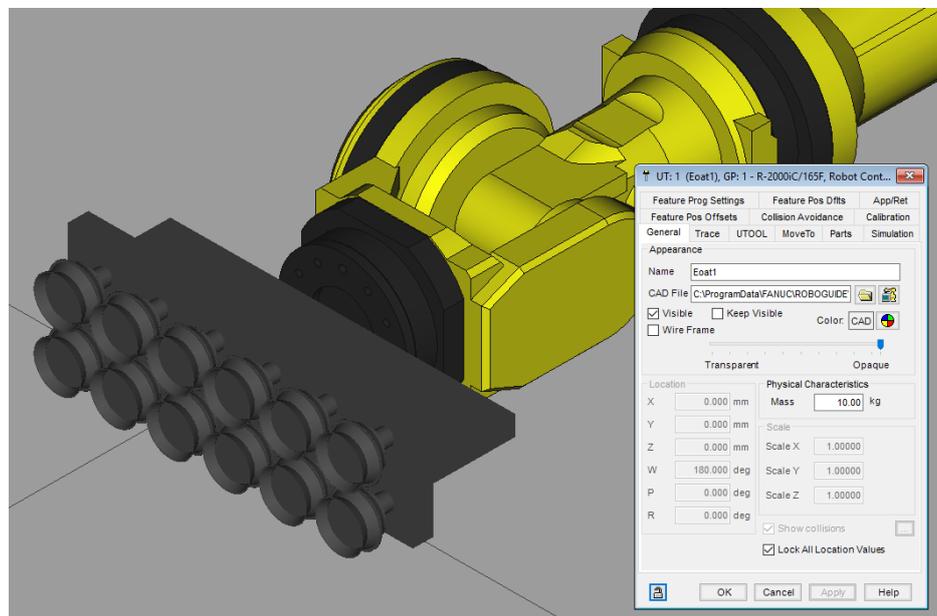


Рис. 12 Корректировка положения инструмента

Далее поставим «галочку» в пункте *Lock All Location Values* (Заблокировать все значения величин), чтобы зафиксировать инструмент, и нажмём кнопку *Apply*.

### 5.2.3. Добавление конвейера из библиотеки

Для добавления в проект конвейера обратимся к дереву проекта. Правой кнопкой мышки кликнем на пункте «*Fixtures*» (Приспособления), выберем пункт «*Add Fixture*» (Добавить приспособление) и кликнем на пункте «*CAD Library*» (Библиотека моделей) (рис. 5.13).

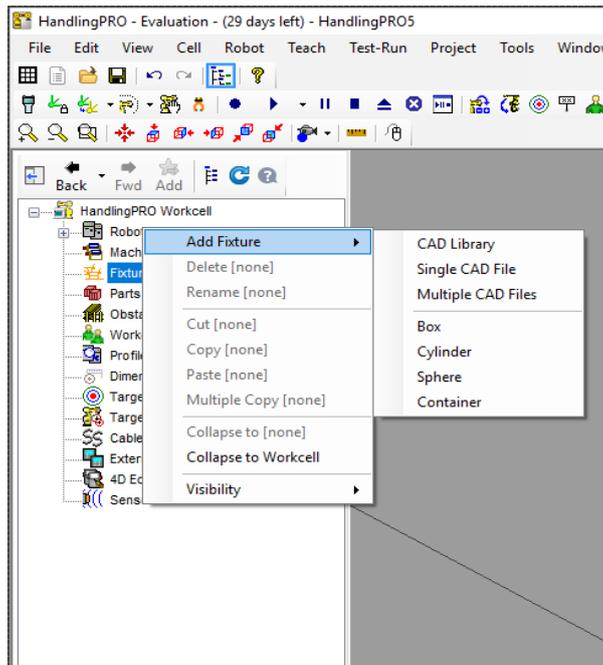


Рис. 5.13 Вызов библиотеки приспособлений

Откроем вкладку «*Conveyor*» (Конвейер) и выберем файл *Conveyor11.iges* (рис. 5.14). Нажмём кнопку ОК.

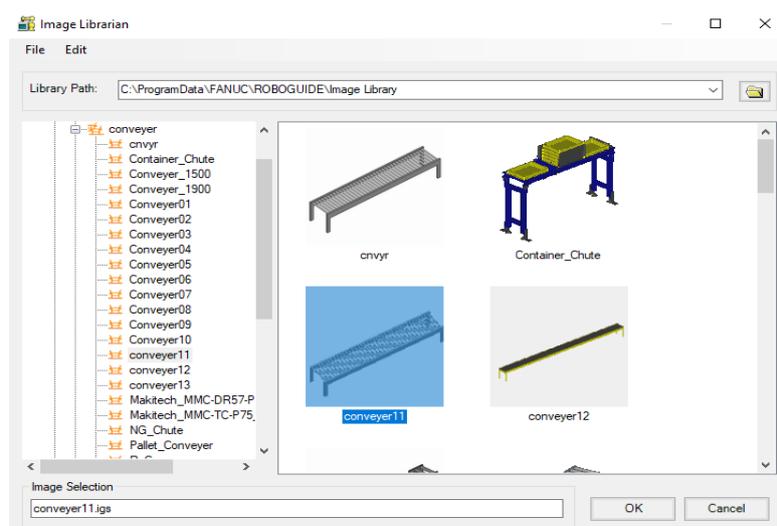


Рис. 5.14 Выбор конвейера

После того как модель конвейера была импортирована в проект, мы можем поместить её в необходимом месте, перемещая

и вращая её за оси координат объекта или введя координаты в соответствующие поля в свойствах объекта вручную. Для облегчения навигации и работы с вновь импортируемыми объектами их рекомендуется переименовывать. Добавленные объекты можно копировать и вставлять в проект, пользуясь стандартными командами *Windows (Copy и Paste)*. Копирование осуществляется в дереве проекта (*Cell Browser*).

#### 5.2.4. Добавление препятствий

Препятствием могут являться следующие объекты:

- контроллеры и шкафы управления;
- защитные ограждения;
- иные объекты и сооружения, ограничивающие рабочее пространство робота.

Препятствие можно добавить в проект из библиотеки или импортировать в формате *IGES*. Для добавления препятствия необходимо в дереве проекта кликнуть правой кнопкой мыши на пункте *Obstacle* и далее по вновь открывающимся вкладкам *Add Obstacle* и *CAD Library* (рис. 5.15) выбрать и импортировать препятствие.

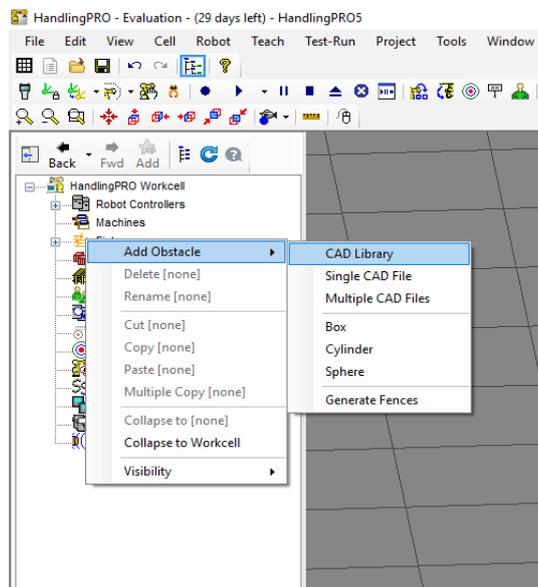


Рис. 5.15 Выбор препятствия

Расстановка, выбор размеров и копирование препятствий выполняется аналогично выбору и копированию конвейера. Необходимо иметь в виду, что расстановку ограждений (*fence*) необходимо выполнять, активировав кнопку в главном меню про-

граммы – *Show/Hide Work Envelope* (рис. 5.16). После её нажатия вокруг робота появляется сфера, ограничивающая его рабочее пространство (рис. 5.17). Ограждения не должны её пересекать.

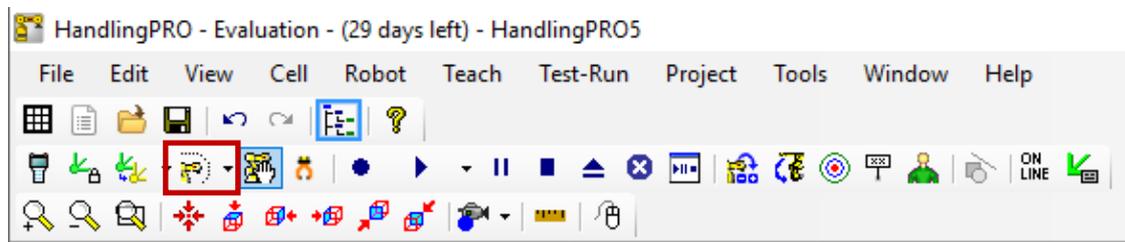


Рис. 5.16 Активация рабочего пространства

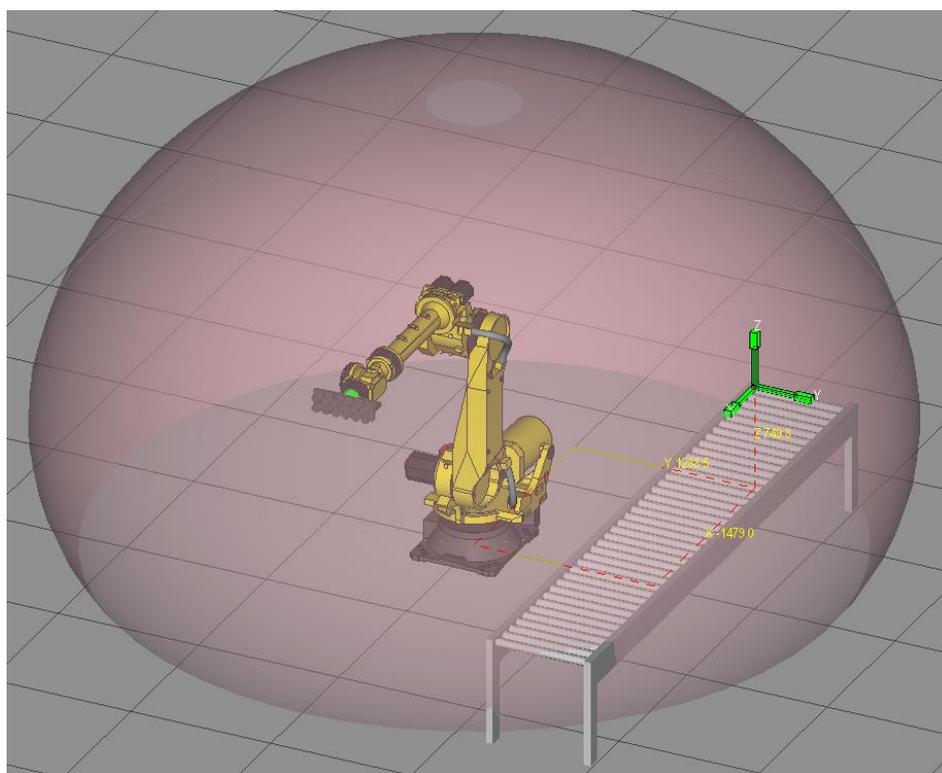


Рис. 5.17 Рабочее пространство робота

На рис. 5.18 представлен пример созданной роботизированной ячейки с одним роботом.

### 5.3. ЗАДАНИЕ

В соответствии с вариантом задания (табл. 5.1) необходимо в программной среде *ROBOGUIDE* построить модель РТК, состоящую из двух роботов, конвейера, ограждений и контроллера. Элементы необходимо расположить таким образом, чтобы конвейер находился внутри рабочего пространства обоих роботов, а

ограждения и контроллер – за его пределами. Тип робота 1 – *ArcTool*, тип робота 2 – *HandlingTool*.

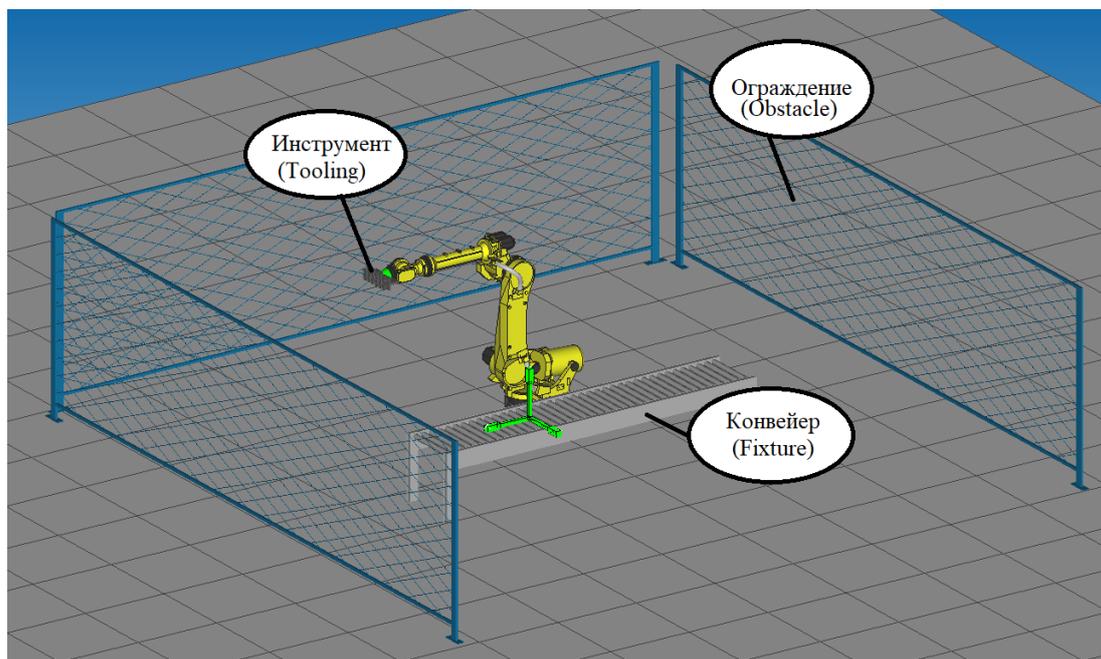


Рис. 5.18 Пример роботизированной ячейки

Таблица 5.1

Варианты заданий

№ вар.	Модель		Инструмент		Приспособление
	робот 1	робот 2	робот 1	робот 2	
1	<i>M-20iA</i>	<i>M-410iC</i>	<i>Precitec_YR30 (cutters)</i>	<i>grippers</i>	<i>Conveyer_1500</i>
2	<i>M-10iA/10M</i>	<i>M-10iA/12</i>	<i>Ultrasonic_kinf e4(cutters)</i>	<i>grippers</i>	<i>Conveyer_1900</i>
3	<i>M-10iA/12</i>	<i>LR Mate 200iD</i>	<i>Pointer(pointers)</i>	<i>grippers</i>	<i>Makitech_MMC-TC-P75_W500_90deg_R900</i>
4	<i>LR Mate 200iB</i>	<i>M-10iA/10M</i>	<i>Precitec_YR30(cutters)</i>	<i>grippers</i>	<i>Conveyer11</i>
5	<i>M-410iC</i>	<i>M-20iA</i>	<i>Ultrasonic_kinf e4(cutters)</i>	<i>grippers</i>	<i>Conveyer_1500</i>

## 5.4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Как в построенную модель РТК добавить ещё одного робота?
2. Как оснастить выбранный робот захватным устройством?
3. По какому принципу расставляется ограждение РТК?

## 5.5. ЛИТЕРАТУРА

1. Курышкин, Н. П. Основы робототехники : учебное пособие для студентов вузов, обучающихся по направлениям подготовки: 220700.62 "Автоматизация технолог. процессов и пр-в", профиль 220701.62 "Автоматизация технолог. процессов и пр-в в машиностроении"; 151900.62 "Конструкт.-технолог. обеспечение машиностроит. пр-в", профиль 151901.62 "Технология машиностроения"; 150700.62 "Машиностроение", профиль 151901.62 "Оборудование и технология свароч. пр-ва" / Н. П. Курышкин ; ФГБОУ ВПО «Кузбас. гос. техн. ун-т им. Т. Ф. Горбачева», Кафедра приклад. механики. – Кемерово : Издательство КузГТУ, 2012. – 168 с. – Доступна электронная версия: <http://library.kuzstu.ru/meto.php?n=90828&type=utchposob:common>
2. Руководство по работе в программной среде *RoboGuide V 6.40*, режим доступа: <http://belfingroup.com/belfingrupp/spravka/skachat/pdf-katalogi-funuc.html> (дата обращения 24.02.2024г.).

## 6. ПРОГРАММИРОВАНИЕ РОБОТА В СРЕДЕ ROBOGUIDE

Цель – приобретение навыков программирования промышленных роботов (ПР) в программной среде *ROBOGUIDE V9*, входящих в состав виртуальной роботизированной ячейки.

С этой целью, используя результаты практического занятия «Создание роботизированной ячейки в программной среде *ROBOGUIDE*», по приведённой ниже методике разрабатывается программа работы двух промышленных роботов *FANUC*.

Работа выполняется на компьютере с загруженным пакетом *ROBOGUIDE V9* и рассчитана на два часа.

### 6.1. МЕТОДИКА РАЗРАБОТКИ ПРОГРАММЫ

Результатом предыдущего практического занятия является виртуальный роботизированный технологический комплекс (РТК), включающий два промышленных робота *FANUC*, конвейер, ограждения, контроллер. Один робот – ПР1 – снабжён захватным инструментом с подвижными губками, а второй – ПР2 – технологическим инструментом.

Программа работы РТК должна обеспечить работу роботов по следующему рабочему циклу:

- конвейер подаёт заготовку в рабочую позицию;
- робот ПР1 берёт заготовку с конвейера и переносит её на рабочий стол;
- робот ПР1 ставит заготовку на рабочий стол и, разжав губки схвата, поворачивается на безопасное от рабочего стола расстояние;
- робот ПР2 подводит инструмент к заготовке и «рисует» на её поверхности замкнутую линию (форма линии определяется вариантом задания);
- по окончании «рисования» робот ПР2 возвращается в исходное положение;
- робот ПР1 берёт с рабочего стола готовую деталь и ставит её обратно на конвейер;
- робот ПР1, разжав губки схвата, возвращается в исходное положение.

Перед разработкой программы работы РТК его состав необходимо дополнить заготовкой. Кроме этого, размеры захватного устройства надо «подогнать» под размеры заготовки и обеспечить губкам схвата подвижность.

### 6.1.1. Добавление заготовки

Добавление заготовки выполняется в дереве обозревателя проекта (окне навигации) *Cell Drowser* (рис. 6.1, а). Для этого необходимо кликнуть правой кнопкой мыши на вкладке *Parts*, а затем в раскрывшемся меню выбрать *Add Part* (добавление детали). В открывшемся подменю из трёх предлагаемых форм заготовки (параллелепипед, цилиндр, шар) необходимо выбрать (*Box*) параллелепипед (рис. 6.1, а). Отметим, что в программе имеется возможность выбора заготовки из библиотеки *CAD Library*, либо импорта её *CAD*-модели в формате *IGES* из любой программы *3D*-моделирования.

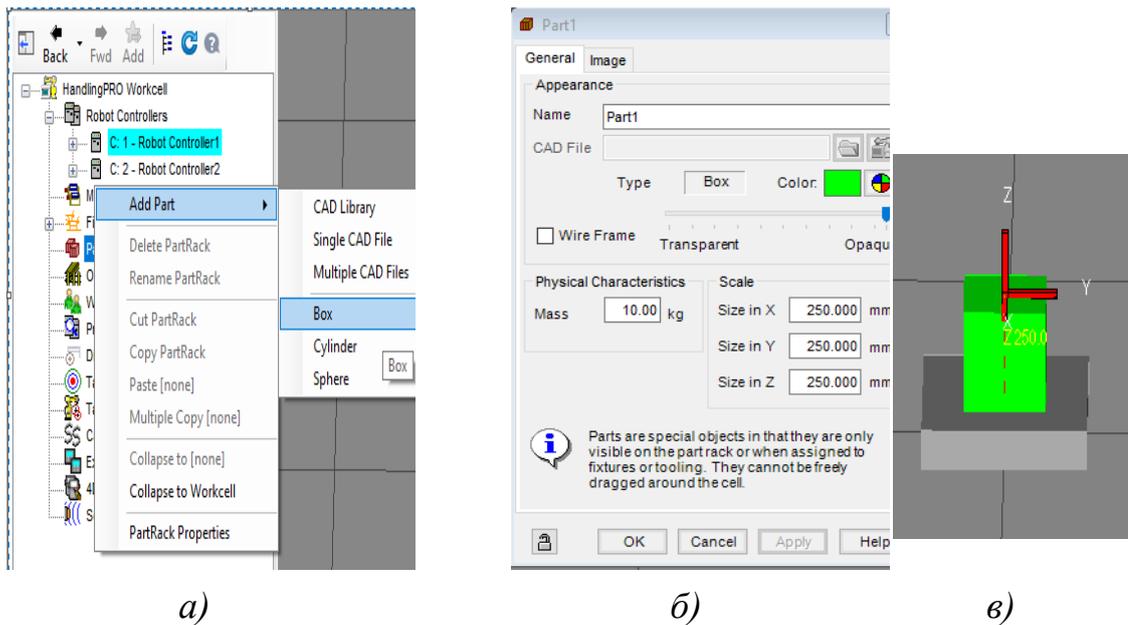


Рис. 6.1 Добавление заготовки

В открывшемся окне свойств детали (рис. 1, б) необходимо ввести её имя латиницей, цвет, размеры и массу. Размеры рекомендуется выбрать, как показано на рис. 6.1, б. Остальные параметры – по предпочтению. Далее нажимаем «Apply» и «OK». Заготовка добавлена (рис. 6.1, в).

### 6.1.2. Добавление и симуляция захватного устройства

Добавление захватного устройства выполняется также в дереве обозревателя проекта. Пройдя по ссылке, как показано на рис. 6.2, а, необходимо открыть свойства инструмента «*Eoat1 Properties*» во вкладке «*UT: 1*». Откроется окно свойств захватного устройства (рис. 6.2, б). В нём необходимо открыть вкладку «*General*».

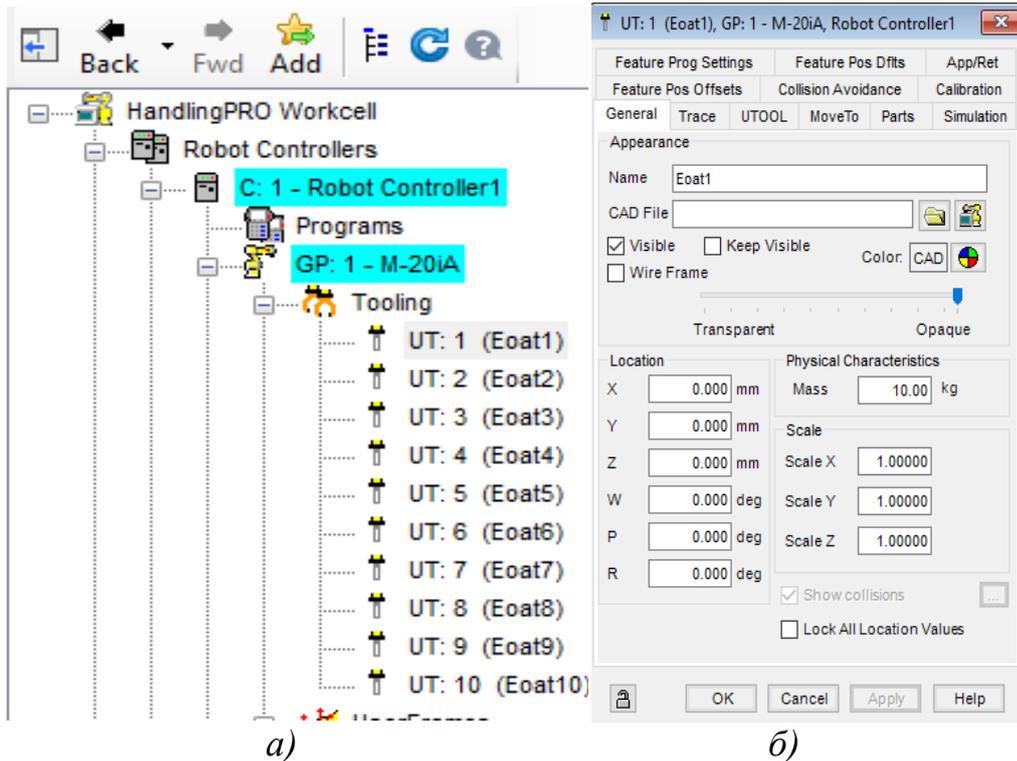
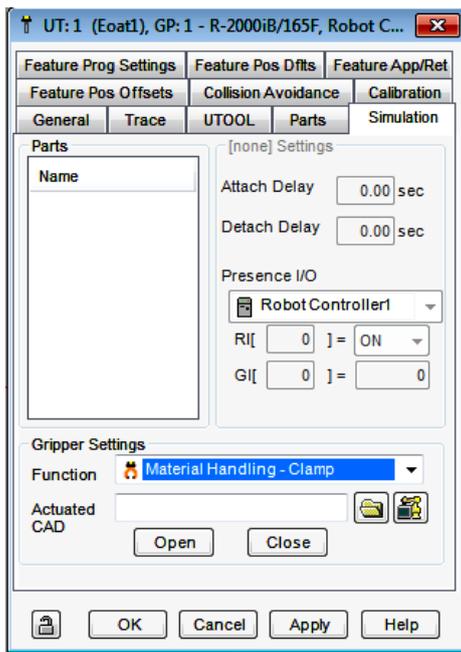


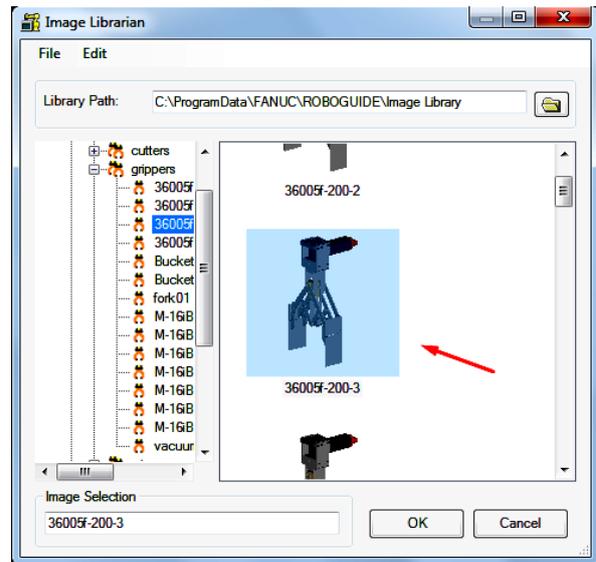
Рис. 6.2 Добавление захвата

В этой вкладке, используя кнопку с изображением открытой папки, выбирается нужное захватное устройство (в открытом состоянии!). Можно присвоить ему имя, используя строку *Name*. Ниже указываются параметры его положения и масштабные размеры. Эти параметры и размеры вводятся под визуальным контролем захватного устройства на экране.

Параметры симуляции захватного устройства выбираются во вкладке его свойств *Simulation* (рис. 6.3, а). В строке *Function* необходимо выбрать «*Material Handling Clamp*», а в строке «*Actuated CAD*» необходимо выбрать захват в закрытом состоянии из нескольких вариантов в зависимости от размеров захватываемого объекта (рис. 6.3, б). Библиотека этих вариантов открывается с использованием кнопки со значком робота на рис. 6.3, а.



a)



б)

Рис. 6.3 Симуляция захвата

Далее во вкладке *UTOOL* свойств инструмента необходимо поставить «галочку» в пункте «*Edit UTOOL*» и переместить рабочую точку *TCP* робота ПР1 в середину между губками захватного устройства (рис. 6.4, а). Затем надо нажать кнопку «*Use Current Triad Location*» (записать текущие координаты) и кнопку «*Apply*» (применить).

Привязка заготовки к захватному устройству выполняется во вкладке «*Parts*». Открыв эту вкладку, необходимо поставить «галочки» напротив нужной заготовки и у надписи «*Edit Part Offset*». После этого «перетаскиванием» заготовки вдоль её осей координат или вводя численные значения этих координат в поле *Part Offset* вкладки «*Parts*», задаётся правильное положение заготовки относительно захвата. Далее нажимается кнопка «*Apply*». Нажатием кнопки «*OK*» окно свойств захватного инструмента закрывается.

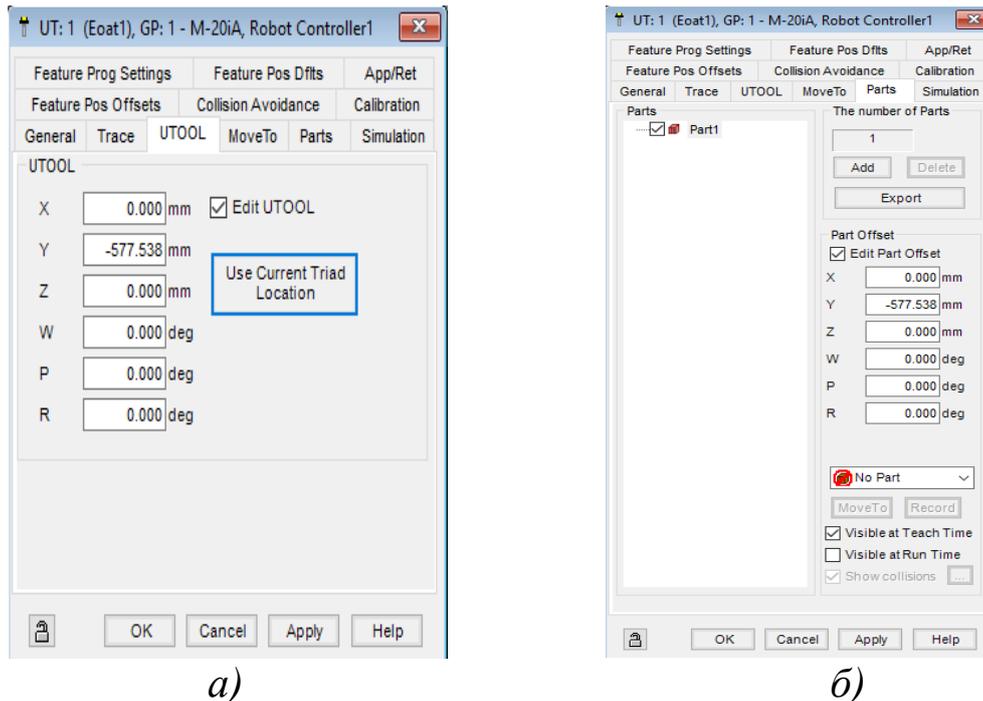


Рис. 6.4 Привязка заготовки к захвату

Примечание: добавление и привязка таких инструментов,

как «*Pointer*» (указатель) или «сварочная горелка» выполняется аналогично добавлению захватного инструмента за исключением заполнения вкладки *Simulation*.

### 6.1.3. Привязка заготовки к устройствам позиционирования

Выбранная заготовка должна быть привязана не только к захватному устройству, но и к устройствам позиционирования – конвейеру и рабочему столу. Для привязки к конвейеру необходимо щёлкнуть двойным кликом мыши на его изображении. Откроется окно свойств конвейера (рис. 6.5). Выбрав вкладку «*Parts*», необходимо поставить

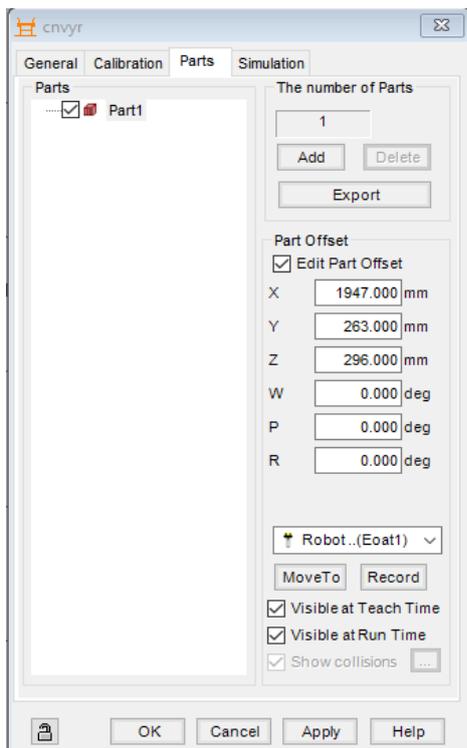


Рис. 6.5 Конвейер

«галочку» напротив нужной детали (заготовки). Далее, поставив «галочку» около надписи «*Edit Part Offset*», нужно отпозиционировать выбранную заготовку относительно конвейера «перетас-

киванием» за оси системы координат или прямым вводом численных значений координат. Затем следует кликнуть на кнопку «Apply».

Заканчивается привязка заготовки выбором параметров во вкладке «Simulation» (рис. 6.6, а).

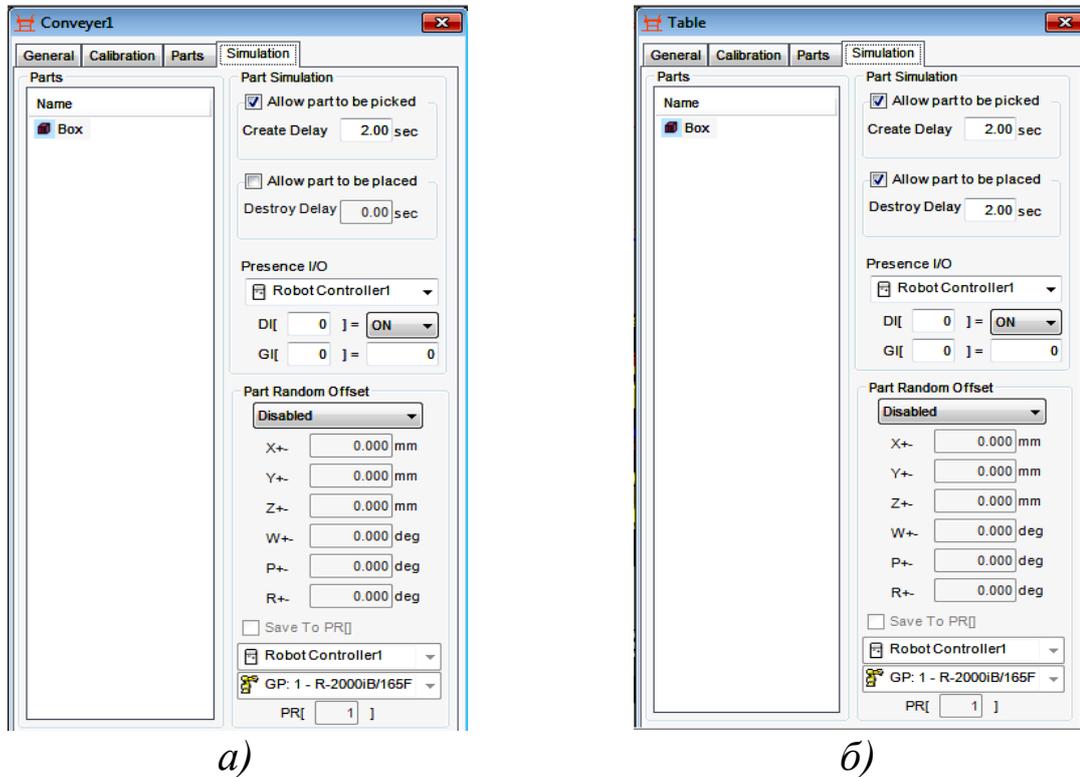


Рис. 6.6 Выбор параметров вкладки «Simulation»

Необходимо поставить «галочку» у надписи «*Allow part to be picked*» и установить нужное время (заготовка будет появляться на конвейере через заданное время).

Так как на этот же конвейер будет устанавливаться готовая деталь в конце цикла, то нужно ещё поставить «галочку» напротив надписи «*Allow part to be placed*» и установить требуемое время (деталь будет исчезать с конвейера через заданное время). Далее нажать последовательно кнопки «Apply» и «OK».

Привязка заготовки к рабочему столу выполняется аналогично. Двойным кликом мыши на изображении рабочего стола появляется окно свойств этого объекта. Во вкладках «Part» и «Simulation» (рис. 6.6, б) свойств рабочего стола указываются необходимые параметры.

#### 6.1.4. Создание траектории и добавление программы

В соответствии с заданием робот ПР2 должен на поверхности заготовки концом своего инструмента пройти по замкнутой траектории, форма которой определяется вариантом задания. В программной среде *ROBOGUIDE* предусмотрено автоматическое генерирование траектории четырёх видов:

- *Edge Line* – краевая линия (рис.6.7, а);
- *Freehand Line* – свободная линия (рис. 6.7, б);
- *Surface Fit Line* – линия на поверхности (рис. 6.7, в);
- *Curve* – кривая (рис. 6.7, г).

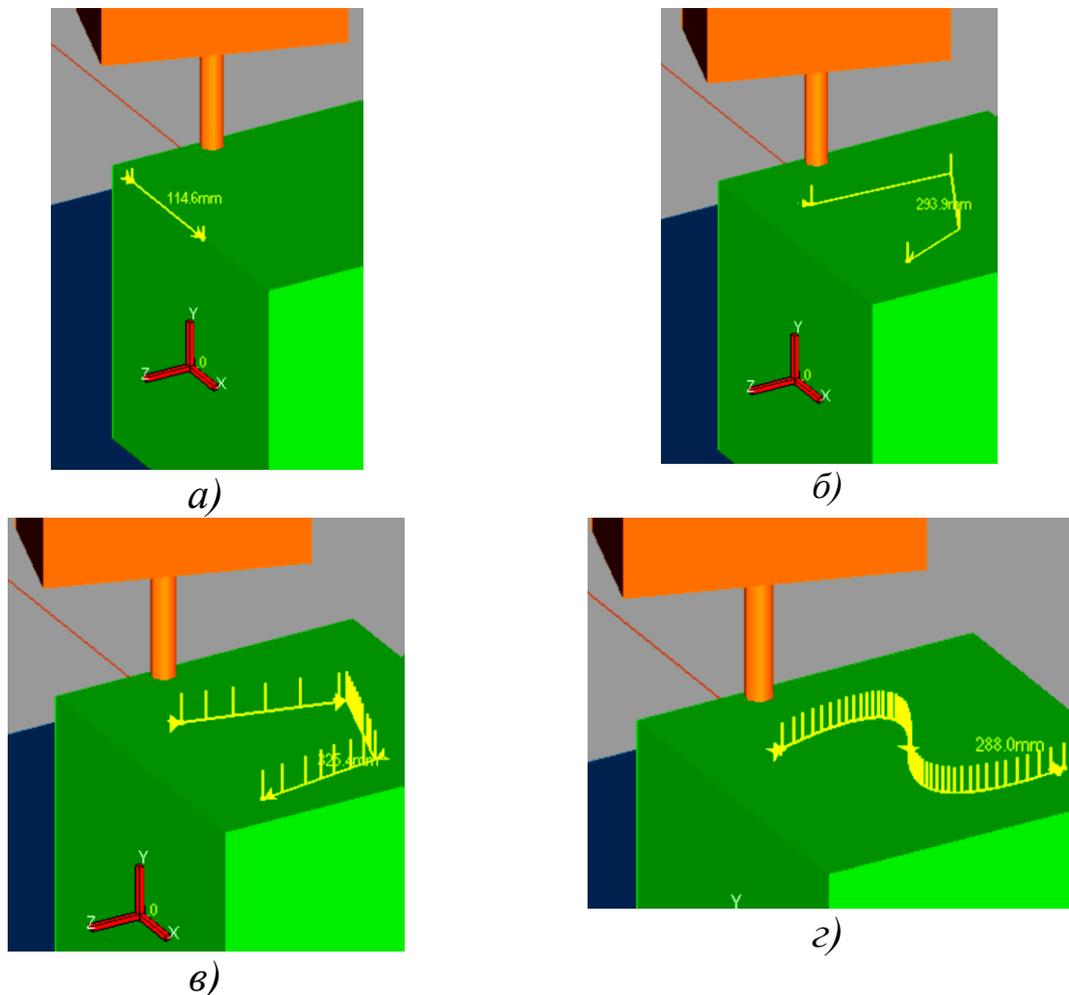


Рис. 6.7 Генерирование траекторий

Программирование траектории начинается с активирования заготовки, находящейся на рабочем столе. Для этого необходимо кликнуть на ней одним щелчком левой кнопки мыши.

После активации заготовки в меню команд необходимо нажать кнопку «*Draw features on parts*» (рис. 6.8). Откроется окно «*CAD-To-Path*» (рис. 6.9). В этом окне



Рис. 6.8 Рисование на детали

можно выбрать генерирование любой из четырёх кривых, показанных на рис. 6.7. В качестве примера рассмотрим алгоритм генерирования траектории, привязанной к грани – краевой линии *Edge Line*.

Генерация начинается с нажатия соответствующей кнопки в окне «*CAD-To-Path*» (рис. 6.9). Далее необходимо подвести курсор на угол ребра заготовки. Кликнув

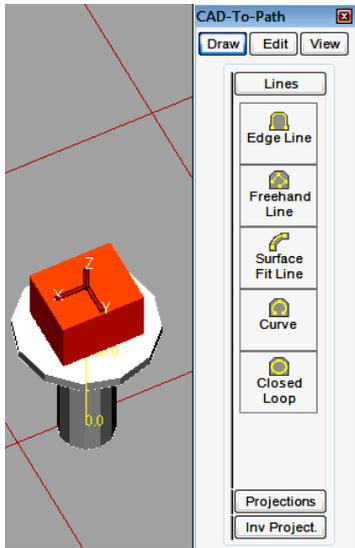


Рис. 6.9 Окно «*CAD-To-Path*»

один раз кнопкой мыши, провести курсор по контуру всех граней заготовки, нажимая один раз на левую кнопку мыши при смене направления грани. После замыкания кривой необходимо дважды кликнуть левой кнопкой мыши. Отобразится пройденное курсором расстояние и появится окно с настройками (рис. 6.10).

В этом окне необходимо открыть вкладку «*General*» и в ней присвоить латиницей имя программе обхода контура граней. Проверить выполненные на этой вкладке настройки по умолчанию.

Далее, необходимо открыть вкладку «*Approach/Retreat*» (точки подвода/отвода инструмента) (рис. 6.11, а) и поставить «галочки» у надписей «*Add approach point*» и «*Add retreat point*». В графе «*Mo Type*» необходимо выбрать «*J*».

Кликнув один раз на левую кнопку мыши при смене направления грани. После замыкания кривой необходимо дважды кликнуть левой кнопкой мыши. Отобразится пройденное курсором расстояние и появится окно с настройками (рис. 6.10).

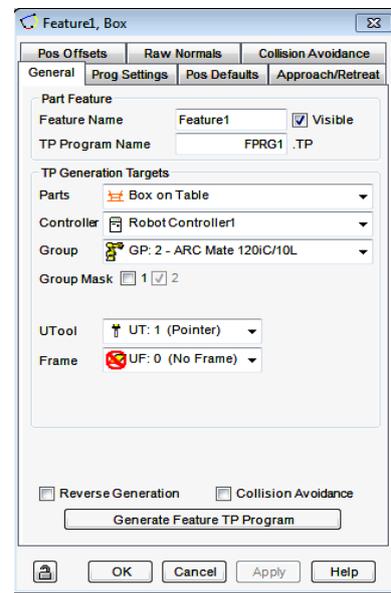
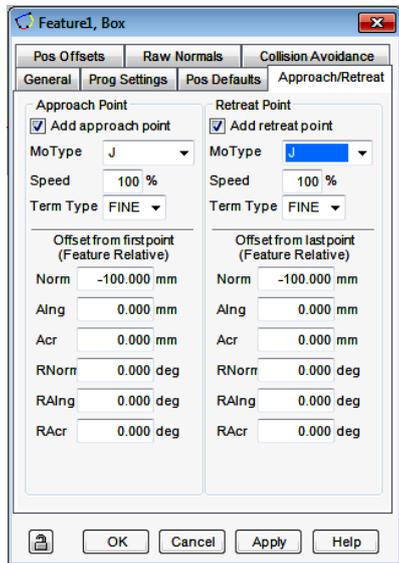
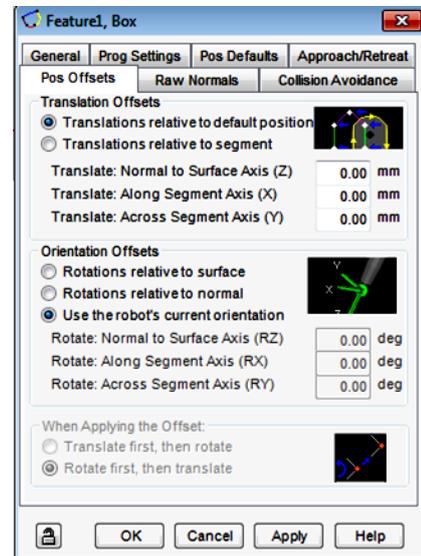


Рис. 6.10 Фиксация траектории

В открытой вкладке «*Pos Offsets*» (рис. 6.11, б) в графе «*Orientation Offsets*» надо выбрать «*Use the robot's current orientation*» (использовать текущую ориентацию робота).



а)



б)

Рис. 6.11 Подвод и отвод инструмента

После этого, вернувшись во вкладку «*General*» (рис. 6.10), необходимо нажать кнопку «*Generate Feature TP Program*» (кнопка находится внизу вкладки).

В открывшемся окне (рис. 6.12) необходимо нажать последовательно кнопки «*Apply*» и «*OK*».

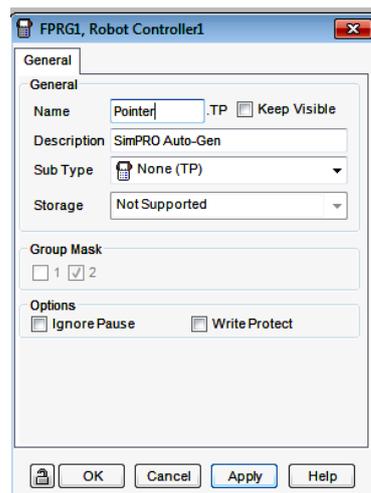


Рис. 6.12 Завершение генерации траектории

Таким образом, программа движения инструмента робота ПР2 по краевой линии с точками подвода и отвода создана.

### 6.1.5. Создание программы моделирования движения

Разработка программы моделирования движения роботов в рабочем цикле начинается в дереве обозревателя проекта (рис. 6.13).

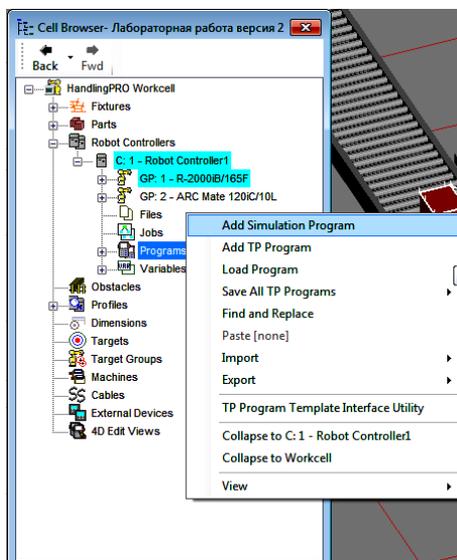


Рис. 6.13 Меню выбора программы

Необходимо перейти во вкладку «*Programs*», нажав на ней правой кнопкой мыши. В открывшемся меню надо выбрать «*Add Simulation Program*». В открывшемся окне, в свою очередь, вводится латиницей имя программы и нажимается кнопка «*OK*».

Добавление команд в программу выполняется в окне, изображённом на рис. 6.14, б).

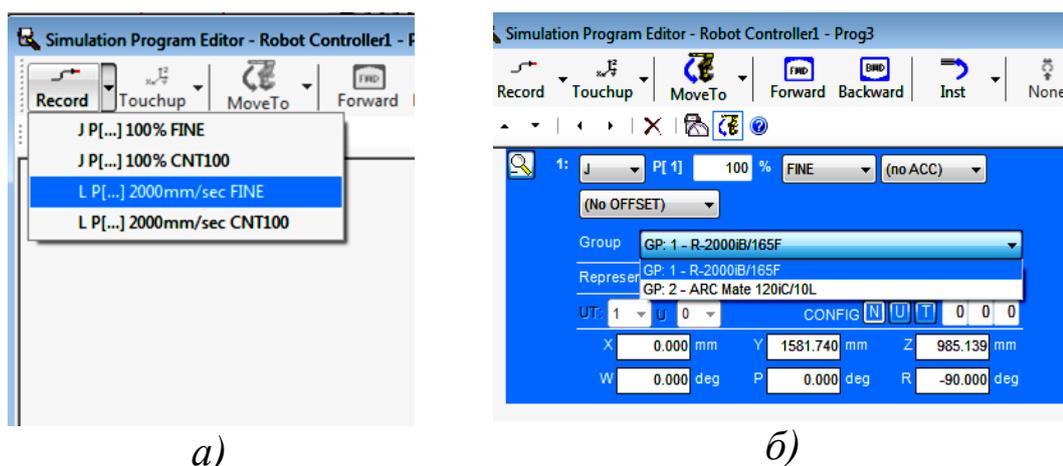


Рис. 6.14 Программирование движения

При нажатии на кнопку «*Record*» будет добавлена точка (рис. 6.14, а). Можно задать её координаты либо путём их прямо-

го ввода в соответствующие поля окна, либо «перетаскиванием» за оси её системы координат.

В строке «*Group*» (рис. 6.14, б) необходимо выбрать нужную группу – робот ПР1 и задать для неё требуемое положение. Если в данный момент будет работать только одна группа (робот), то в других группах всё оставляется без изменений.

Кнопка «*Move To*» (перейти к ...) (рис. 6.15, а) позволяет мгновенно переместить захват манипулятора к выбранному фиксирующему устройству (*Fixtures*) (конвейеру, рабочему столу и т.п.).

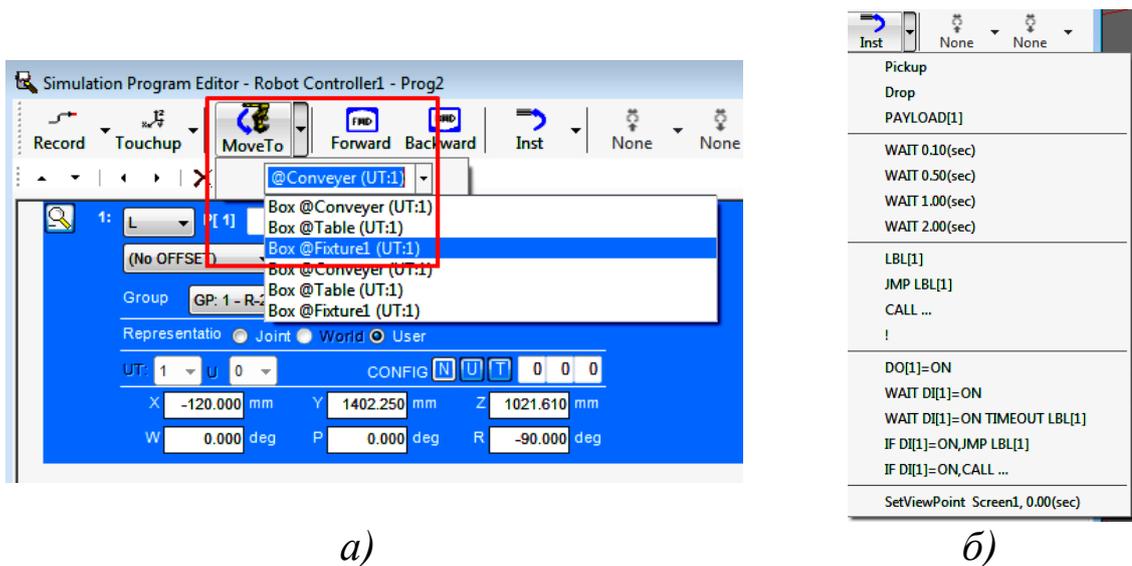


Рис. 6.15 Функциональные клавиши и инструкции

Кнопка «*Inst*» (рис. 6.15, б) открывает список команд, из которого для выполнения практического задания понадобятся следующие:

- «*Pickup*» – команда добавления подпрограммы взятия заготовки; после её исполнения заготовка исчезнет с выбранного фиксирующего устройства (*Fixtures*) (рис. 6.16, а);
- «*Drop*» – команда добавления подпрограммы перемещения заготовки; после её исполнения заготовка появится на выбранном фиксирующем устройстве (*Fixtures*) (рис. 6.16, б);
- «*Call ...*» – эта команда позволяет добавить существующую подпрограмму (например, подпрограмму «рисования» на поверхности заготовки роботом ПР2) в данную главную программу (рис. 6.16, в).

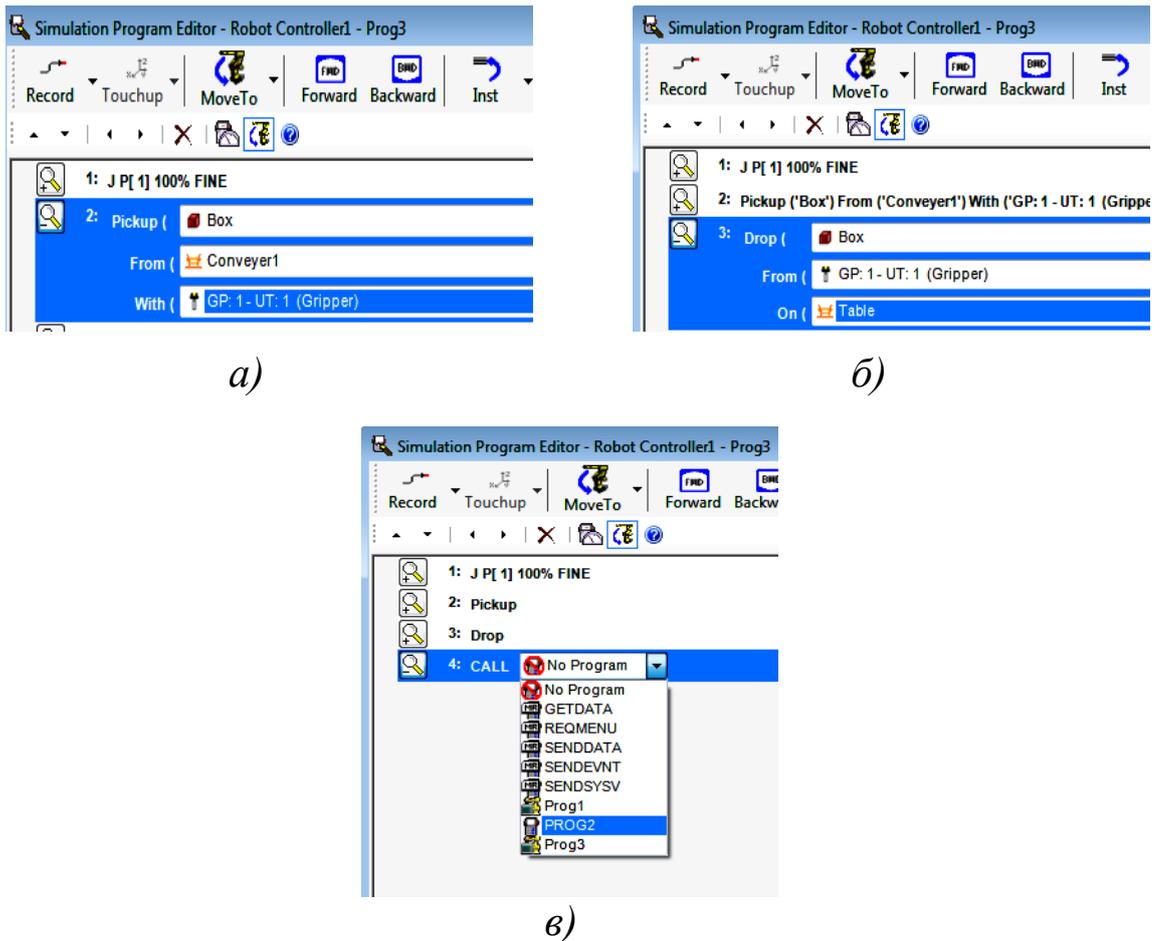


Рис. 6.16 Добавление подпрограмм «взятия», «оставления» и «рисования»

### 6.1.6. Программирование движения роботов

Добавление команд движения в главную программу выполняется также в окне, изображённом на рис. 6.14, а одним нажатием кнопки «Record» при каждом изменении движения. Изменение положения захвата выполняется нажатием на соответствующей цветной оси системы координат точки ТСР правой кнопкой мыши и дальнейшем «перетаскиванием» в нужное положение, как показано на рис. 6.17. После каждого перетаскивания необходимо нажимать кнопку «Record». Чем чаще будет нажиматься эта кнопка, тем точнее будет программироваться траектория движения захвата. При каждом нажатии кнопки «Record» мы наблюдаем синюю линию траектории перемещения захвата.

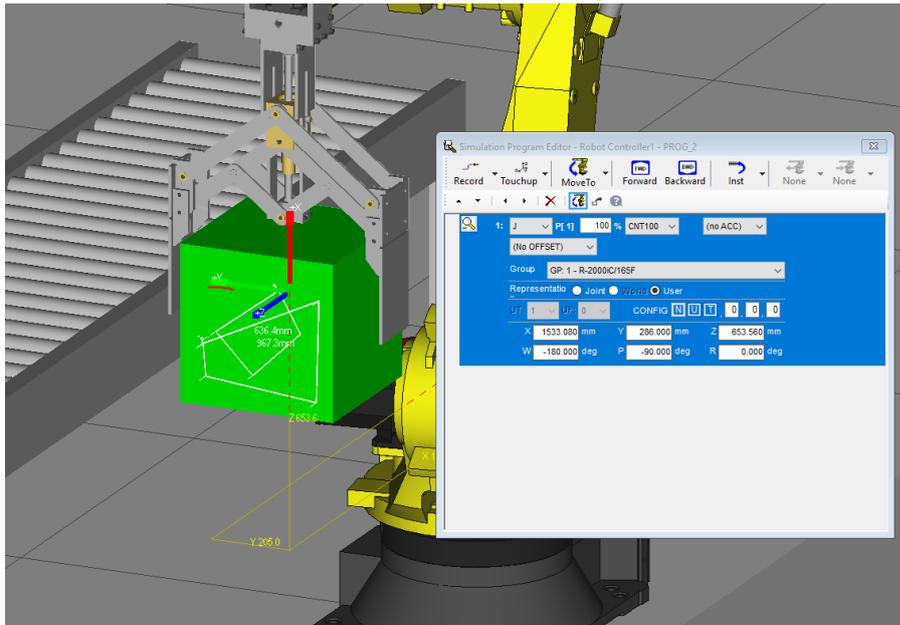


Рис. 6.17 Программирование движения

#### 4.1.1 Запуск программы

Запуск программы может производиться либо с панели инструментов (1 на рис. 6.18, а), либо из окна «Run Panel» (рис. 6.18, б), которое вызывается нажатием на соответствующий значок 2 панели инструментов.

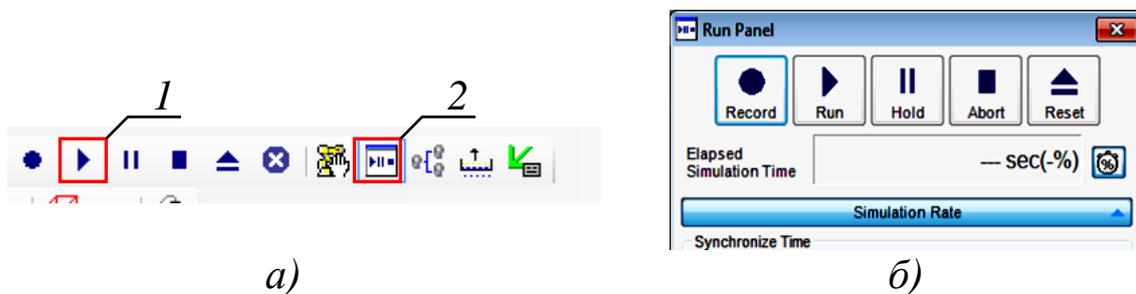


Рис. 6.18 Запуск программы

## 6.2. ЗАДАНИЕ

В соответствии с вариантом кривой, которую должен «нарисовать» робот ПР2 на поверхности заготовки, помещённой на рабочий стол роботом ПР1, разработать программу работы РТК, сформированного на предыдущем практическом занятии. Форма кривой выдаётся преподавателем персонально каждому обучающемуся.

Задание выполняется в программной среде *ROBOGUIDE V9*. Результат работы показывается преподавателю в виде демонстрационного ролика в формате *AVI*.

### 6.3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Как запрограммировать захват заготовки подвижными губками схвата?
2. Какие четыре вида кривых можно автоматически генерировать в среде *ROBOGUIDE*?
3. Как запустить разработанную программу работы РТК в действие и записать демонстрационный ролик?

### 6.4. ЛИТЕРАТУРА

1. Курышкин, Н. П. Основы робототехники : учебное пособие для студентов вузов, обучающихся по направлениям подготовки: 220700.62 "Автоматизация технолог. процессов и пр-в", профиль 220701.62 "Автоматизация технолог. процессов и пр-в в машиностроении"; 151900.62 "Конструкт.-технолог. обеспечение машиностроит. пр-в", профиль 151901.62 "Технология машиностроения"; 150700.62 "Машиностроение", профиль 151901.62 "Оборудование и технология свароч. пр-ва" / Н. П. Курышкин ; ФГБОУ ВПО «Кузбас. гос. техн. ун-т им. Т. Ф. Горбачева», Кафедра приклад. механики. – Кемерово : Издательство КузГТУ, 2012. – 168 с.  
– Доступна электронная версия:  
<http://library.kuzstu.ru/meto.php?n=90828&type=utchposob:common>
2. Руководство по работе в программной среде *ROBOGUIDE V 6.40*, режим доступа: <http://belfingroup.com/belfingrupp/spravka/skachat/pdf-katalogi-funuc.html> (дата обращения 24.02.2024г.).

## 7. РАЗРАБОТКА ПРОЕКТА СВАРОЧНОГО РТК С ДВУХОСЕВЫМ ПОЗИЦИОНЕРОМ В ПРОГРАММНОЙ СРЕДЕ ROBOGUIDE

Цель – приобретение навыков компьютерного проектирования сварочного роботизированного технологического комплекса (РТК) с промышленным роботом *FANUC* и двухосевым позиционером оригинальной конструкции в программной среде *ROBOGUIDE*.

С этой целью необходимо сформировать сварочный РТК в составе сварочного робота *FANUC* и двухосевого позиционера оригинальной конструкции. Все подвижные и неподвижные части позиционера должны быть предварительно спроектированы в какой-либо среде САПР (например, Компас 3D) и импортированы в *ROBOGUIDE* отдельными файлами с расширением *.stl* или *iges*. Имена файлов должны быть набраны латиницей. Допускается использование цифр. В данной работе эти файлы выдаст преподаватель.

### 7.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ

Проект будет включать две группы движения. В первой группе – сварочный робот, во второй группе – двухосевой позиционер оригинальной конструкции. Позиционер будет управляться контроллером робота. Дополнительные оси робота (7 и 8) будут осями позиционера. Чтобы обеспечить это условие, необходимо корректно сформировать проект.

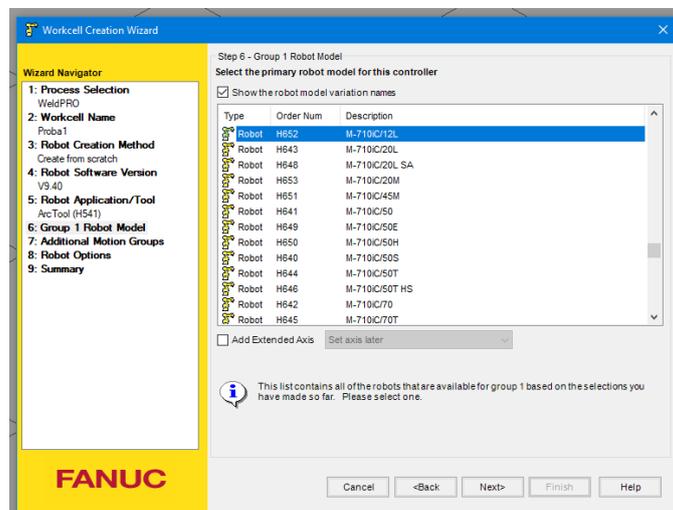


Рис. 7.1 Шаг 6 – выбор модели сварочного робота

Первые шесть шагов формирования проекта – типовые, включая выбор модели сварочного робота (рис. 7.1). На седьмом шаге формирования проекта необходимо выбрать позиционер. Так как в библиотеке *ROBOGUIDE* нет нашего позиционера, то выбираем **2-Axes Servo Positioner (500 kg)** неопределённой конструкции (**Undefined**) с порядковым номером **H871** (рис. 7.2). Далее помещаем этот позиционер в строку 2 нижней таблицы, нажав клавишу **>>** (рис. 7.3).

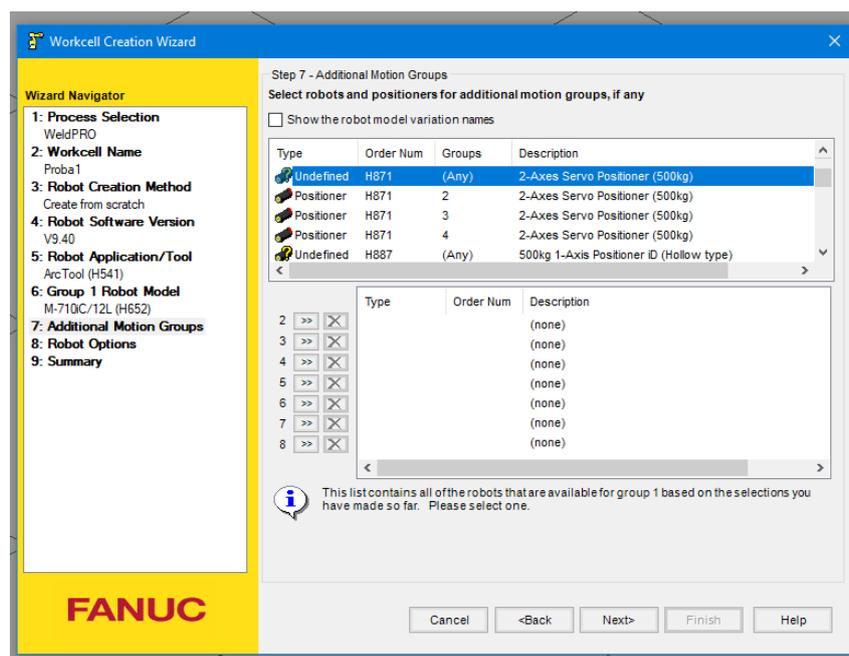


Рис. 7.2 Выбор позиционера

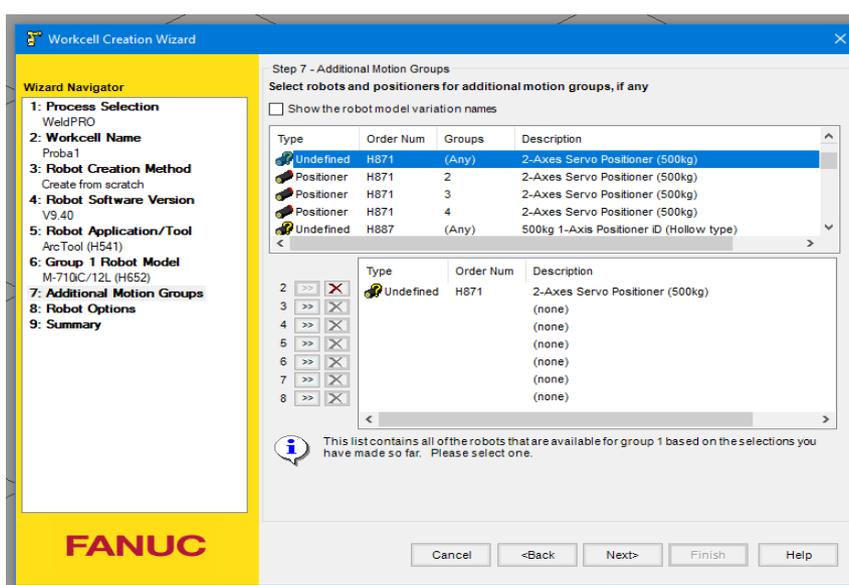


Рис. 7.3 Фиксация позиционера

На восьмом и девятом шаге формирования проекта ничего не добавляем и не меняем, нажимая последовательно клавиши **Next>** и **Finish**. Далее в процессе формирования проекта на экране выносного пульта управления роботом будут появляться вопросы системы по выбору параметров робота и позиционера. Необходимо корректно ответить на эти вопросы согласно рис. 7.4, 7.5, а и 7.5, б.

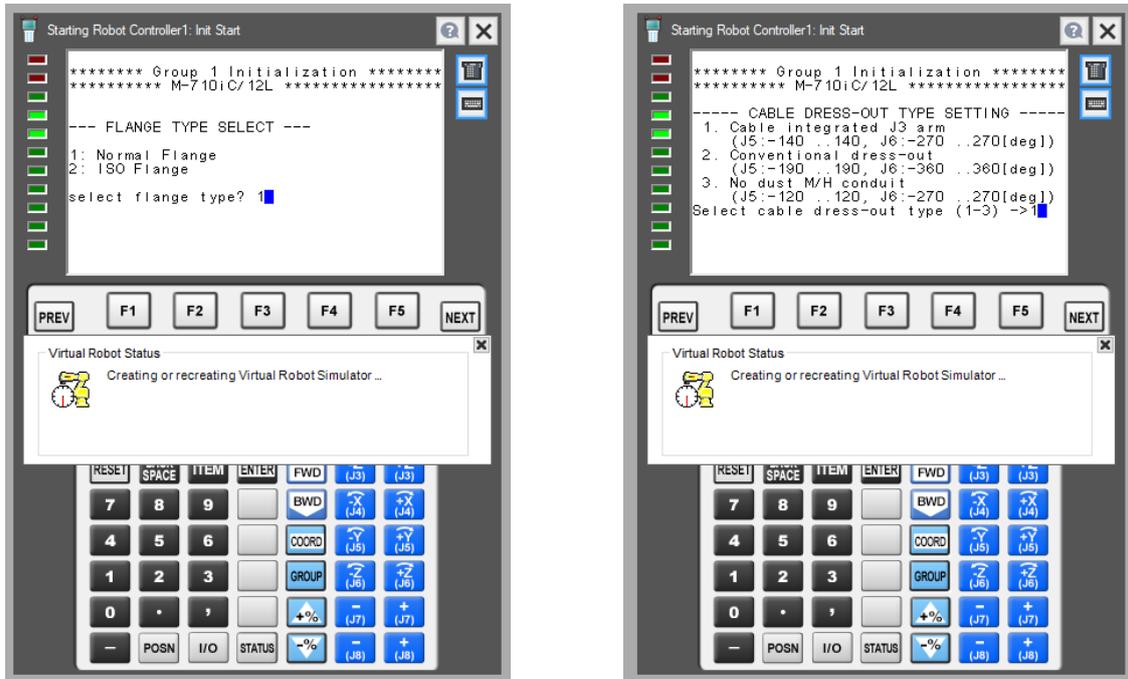


Рис. 7.4 Параметры робота

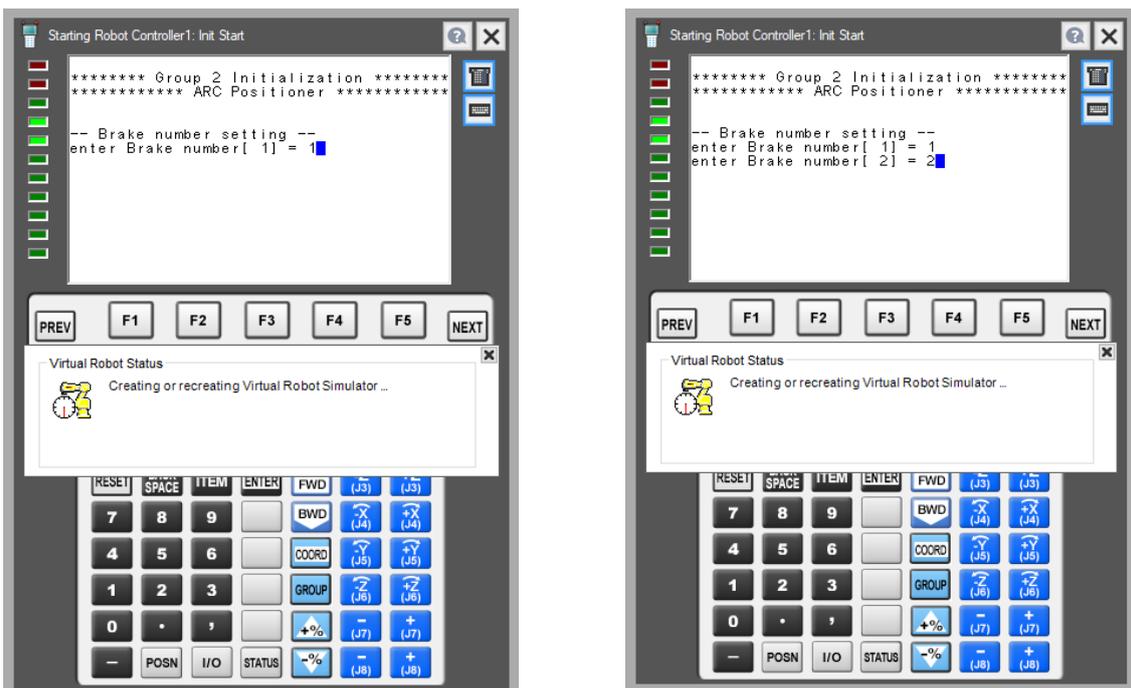


Рис. 7.5, а Параметры позиционера

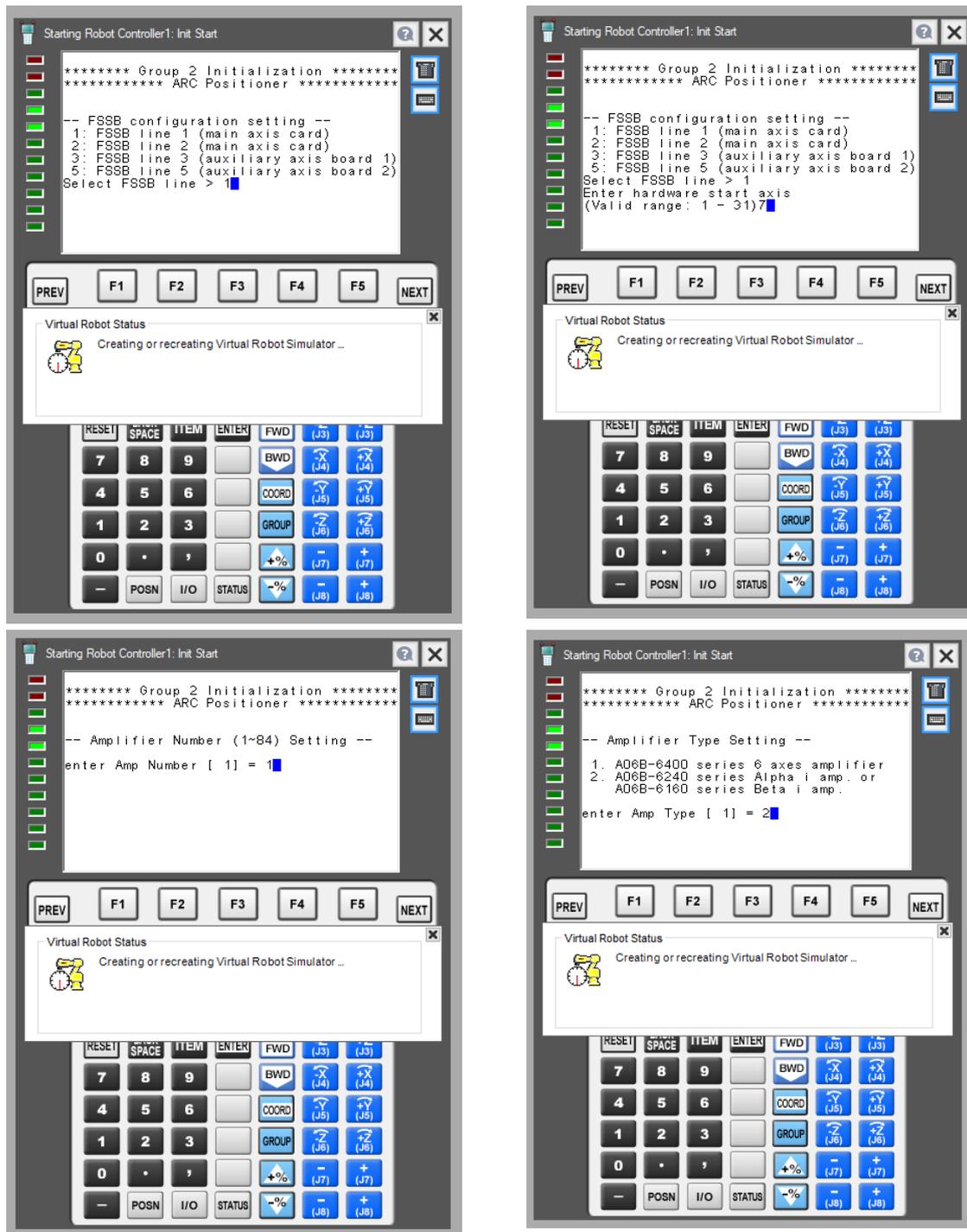


Рис. 7.5, б Параметры позиционера

После завершения первого этапа формирования проекта на экране появится изображение робота. Необходимо проверить отсутствие ошибок при введении параметров проекта. Для этого необходимо активировать в главном меню модель выносного пульта управления (рис. 7.6) и нажать серую клавишу **POSN** справа от основной клавиатуры.

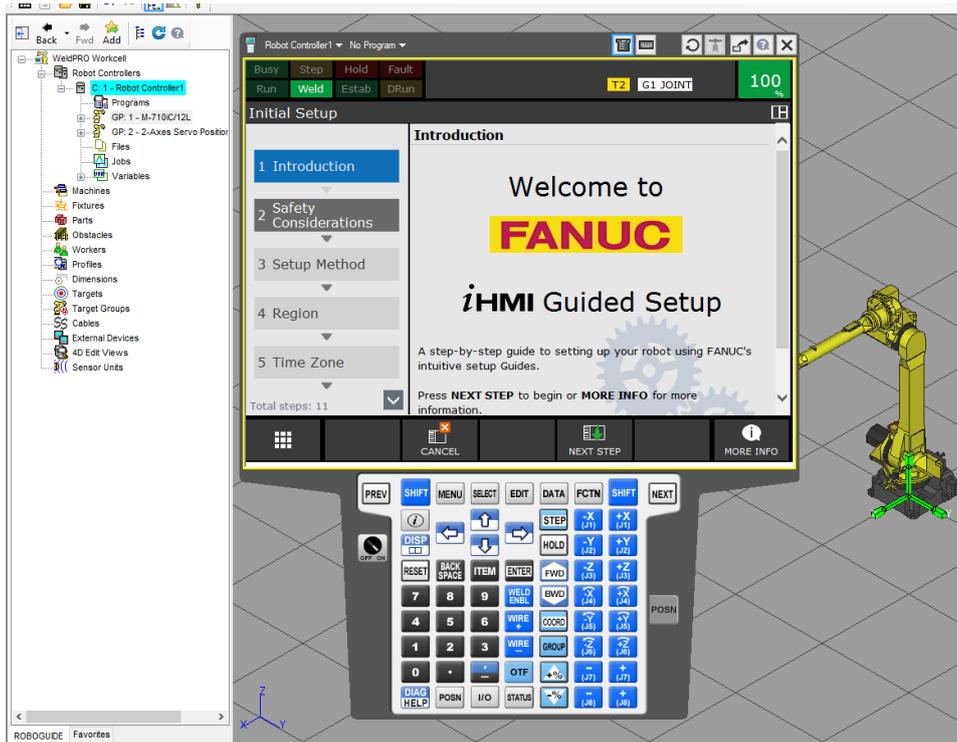


Рис. 7.6 Модель выносного пульта управления

Если в появившемся окне в строке **Group** (рис. 7.7) появилась возможность выбора **GR 2 -2-Axes Servo Positioner (500kg)** и на экране появилась информация об осях позиционера, как на рис. 7.7, то начало формирования проекта выполнено корректно.

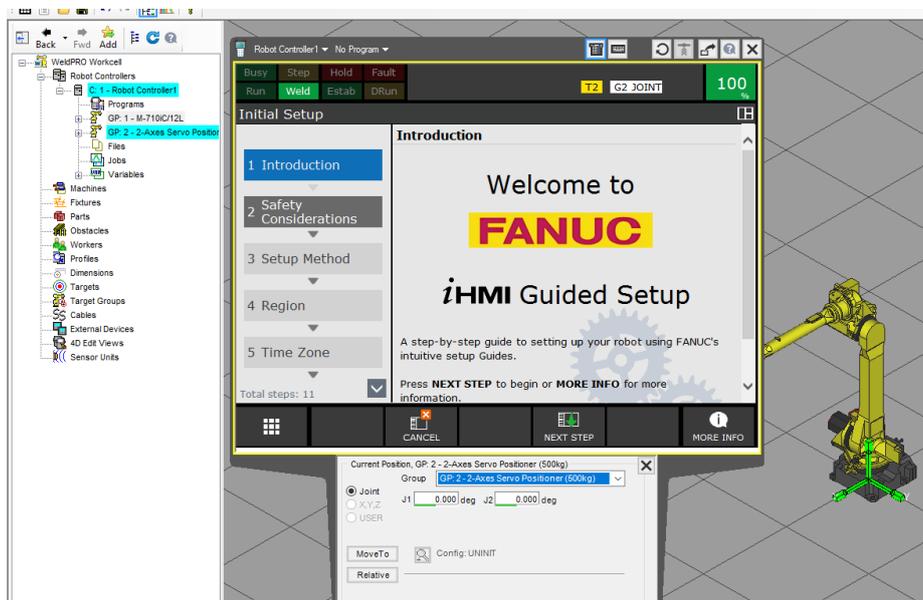


Рис. 7.7 Проверка корректности формирования проекта

На втором и последнем этапе формирования проекта оригинальный позиционер (рис. 7.8) импортируется в проект и «оживляется» в нём.

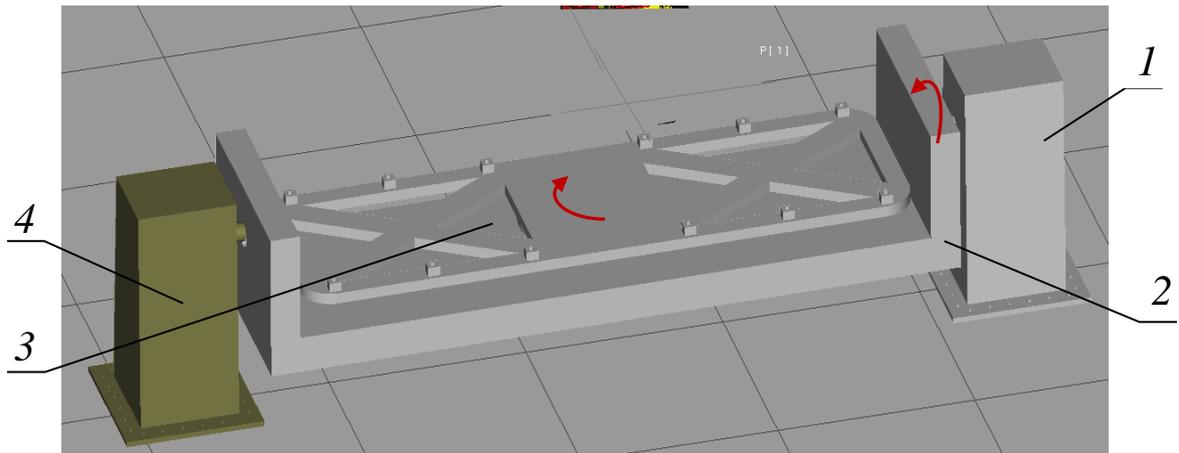


Рис. 7.8 Общий вид позиционера

Позиционер состоит из трёх отдельно спроектированных функциональных узлов: 1 и 4 – стойка, 2 – станина, 3 – стол. В ведущей стойке 1 будет смонтирован двигатель, вращающий станину вокруг горизонтальной оси. В центре станины будет смонтирован двигатель, вращающий стол относительно станины вокруг вертикальной оси. Стойка 4 является поддерживающей и привода не содержит. CAD-файлы функциональных узлов: 1 и 4 – *Stoyka.stl*, 2 – *Stanina.stl*, и 3 – *Stol.stl*.

Импортирование подвижного оборудования (станков, позиционеров и т.п.) выполняется во вкладке **Machines** дерева проекта (рис. 7.9). Важно отметить, что файлы позиционера импортируются в проект *последовательно* по мере необходимости.

После импортирования CAD-файла **Stoyka.stl** на сцене появляется изображение стойки с привязанной к ней системой координат. Используя оси этой системы, стойка позиционируется на своём месте. Далее в дереве проекта добавляем к стойке (строка **Stoyka**) подвижный узел – станину через **Add link** (рис. 7.10): **Machines** → **Stoyka** → **Add link** → **CAD File** → импортируем файл *Stanina.stl*.

Далее в появившемся окне свойств подвижного элемента (**Link1, Stoyka**) во вкладке **General** ставим галочку в квадрате **Edit Axis Origin** (рис. 7.11). На стойке появится изображение

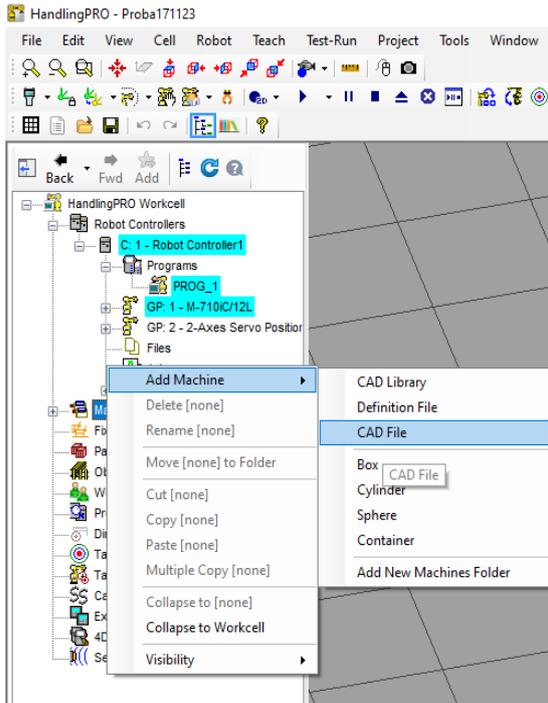


Рис. 7.9 Импортирование CAD-файлов

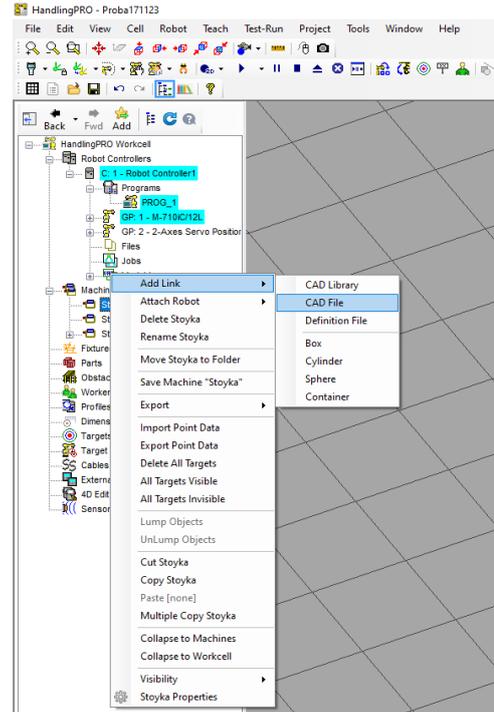


Рис. 7.10 Импортирование подвижных элементов

двигателя со своей системой координат. Используя оси этой системы, необходимо сориентировать этот двигатель так, чтобы его ось **Z** совпала с осью вращения станины. При этом изображение двигателя скроется внутри стойки (рис. 7.12). Во вкладке **Motion** выбрать параметры, как показано на рис. 7.13, и нажать клавишу **Apply**.

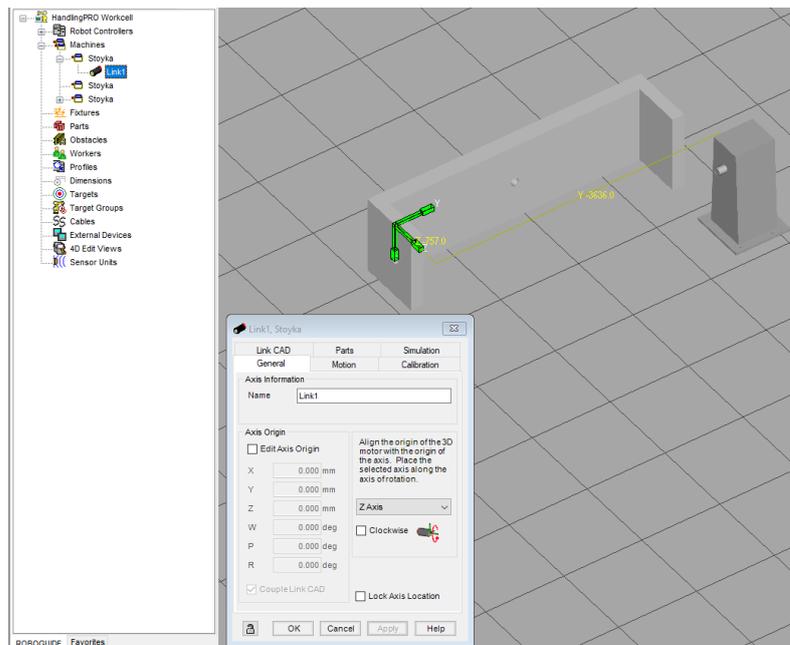


Рис. 7.11 Редактирование подвижного элемента

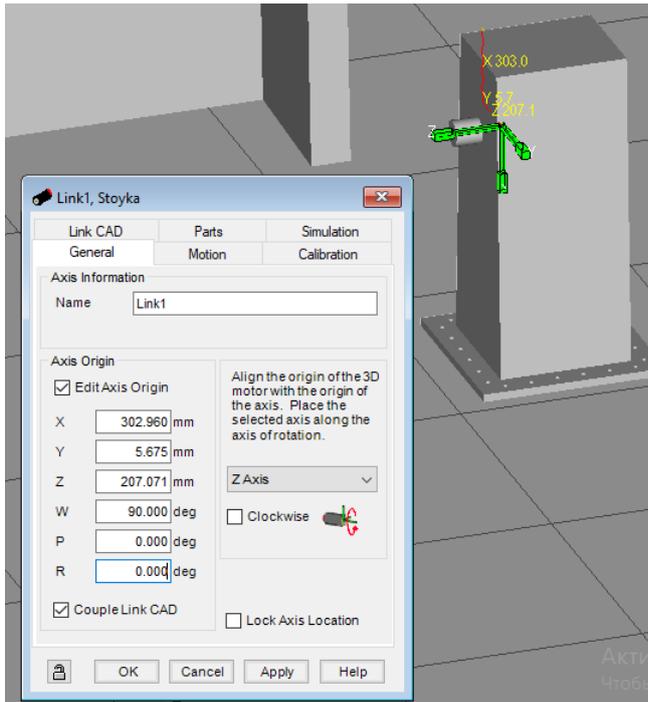


Рис. 7.12 Позиционирование двигателя стойки

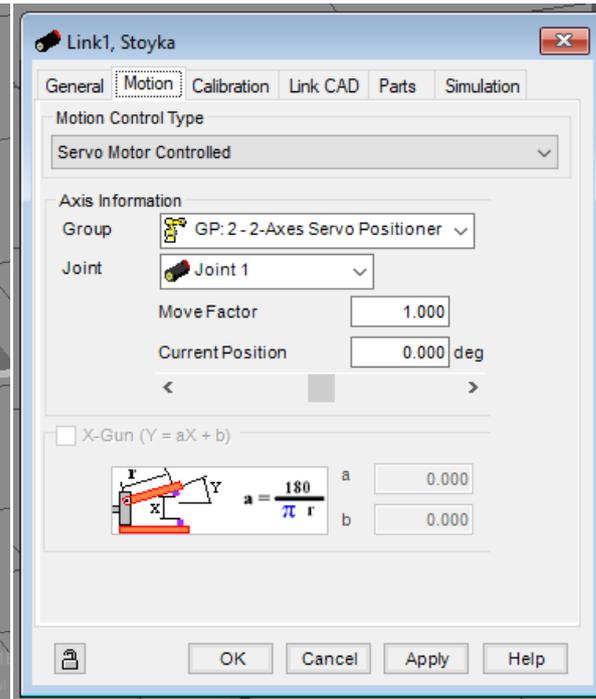


Рис. 7.13 Параметры вкладки **Motion**

Далее, активировав систему координат станины нажатием на её изображении левой кнопкой мыши, сориентировать её относительно стойки, как показано на рис. 7.8.

Для прикрепления второго подвижного элемента – стола – к станине в дереве проекта импортируем CAD-файл *Stol.stl* по схеме как показано на рис. 7.14: **Machines** → **Stoyka** → **Link1** → **Add link** → **CAD File** → импортируем файл *Stol.stl*.

На рис. 7.15 показан результат импортирования файла *Stol.stl* с окном свойств этого объекта. В строке **Name** необходимо скорректировать имя – **Link2** и поставить галочку в квадрате **Edit Axis Origin**.

На станине появится изображение двигателя со своей системой координат. Используя оси этой системы необходимо сориентировать этот двигатель так, чтобы его ось **Z** совпала с осью вращения стола на станине (рис. 7.16) и нажать клавишу **Apply**.

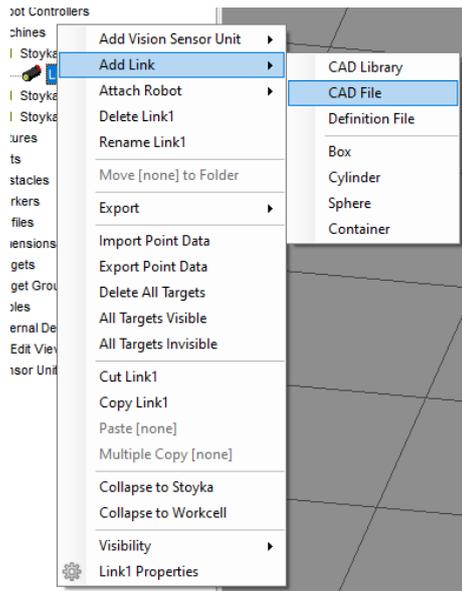


Рис. 7.14 Импортiroвание CAD-файла *Stol.stl*

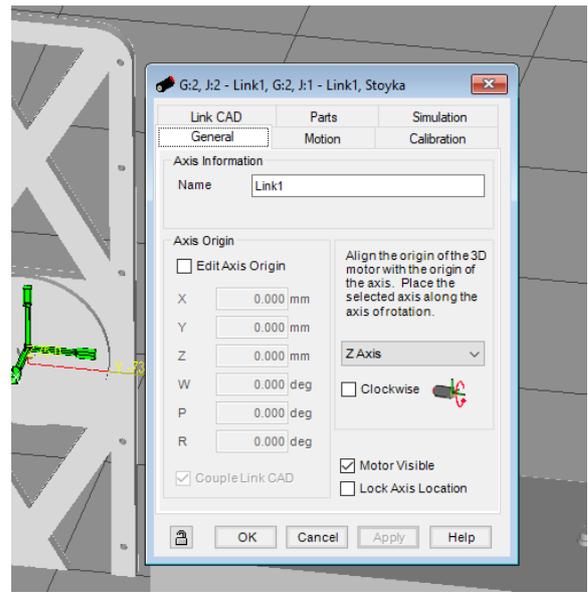


Рис. 7.15 Результат импортирования CAD-файла *Stol.stl*

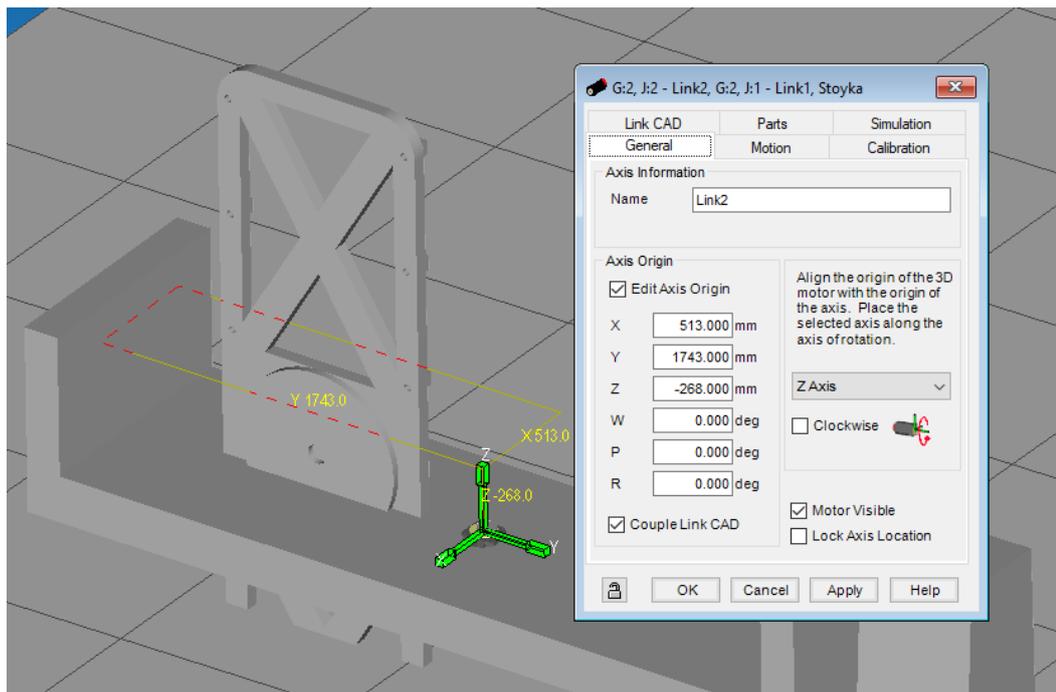
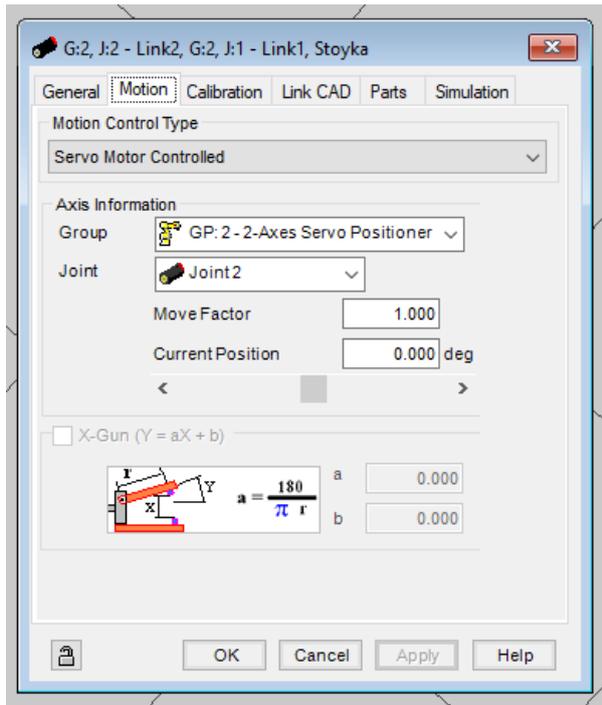
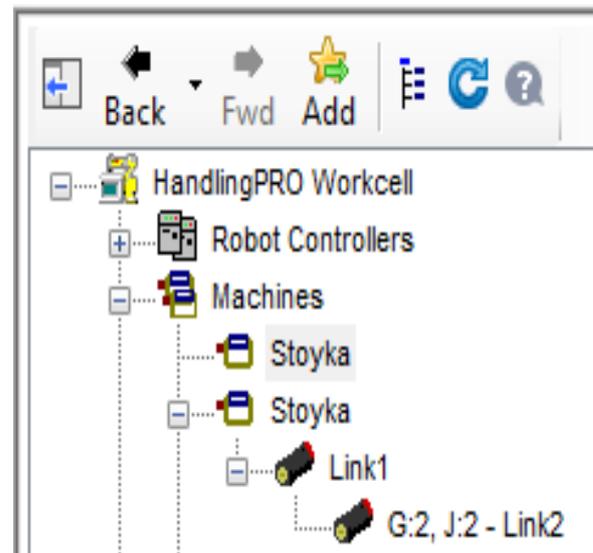


Рис. 7.16 Позиционирование двигателя стола

Далее, активировав систему координат стола нажатием на его изображении левой кнопкой мыши, сориентировать её относительно стойки, как показано на рис. 7.8.

В завершении позиционирования стола необходимо проверить содержимое вкладки **Motion** в окне свойств этого объекта. Содержимое вкладки должно соответствовать рис. 7.17.

Конечным шагом формирования двухосевого позиционера является добавление поддерживающей стойки. Она импортируется файлом *Stoyka.stl* во вкладке **Machines** дерева проекта. Далее, используя оси системы координат стойки, последняя ориентируется, как показано на рис. 7.8. В результате выполненных действий по формированию подвижных элементов позиционера в дереве проекта вкладка **Machines**, должна выглядеть, как показано на рис. 7.18.

Рис. 7.17 Вкладка **Motion**Рис. 7.18 Вкладка **Machines**

На этом формирование двухосевого позиционера оригинальной конструкции заканчивается. Проверка корректности формирования выполняется введением пробных углов поворота осей **J1** и **J2** в соответствующие поля пульта ручного управления (рис. 7.7). После нажатия клавиши **Enter** позиционер должен отработать эти углы.

## 7.2. ЗАДАНИЕ

В программной среде *ROBOGUIDE* сформировать проект сварочного РТК, состоящего из сварочного робота *FANUC M-710iC/12L* и двухосевого позиционера оригинальной конструкции. Позиционер должен управляться контроллером робота.

*CAD*-модели функциональных узлов позиционера получить у преподавателя.

Составить программу работы РТК, включающую одно движение робота из домашнего положения, вращение станины позиционера на 90 градусов и возвращение в исходное положение, далее вращение стола относительно станины на 90 градусов и его возвращение в исходное положение. Завершить программу возвращением робота в домашнее положение.

Продемонстрировать преподавателю работу РТК по составленной программе.

### 7.3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Как проверить, что позиционер управляется контроллером робота?
2. Какое расширение должны иметь *CAD*-файлы функциональных узлов позиционера или другого динамического объекта?
3. В какой вкладке свойств подвижного узла позиционера ориентируется его двигатель?
4. Чем отличается формирование РТК со стандартным позиционером *FANUC* от формирования РТК с позиционером оригинальной конструкции?
5. В какой вкладке дерева проекта импортируются дополнительные динамические объекты (позиционеры, станки и т.п.)?

### 7.4. ЛИТЕРАТУРА

1. Официальный сайт компании *FANUC*, режим доступа [www.fanuc.eu/ru/ru](http://www.fanuc.eu/ru/ru) . (дата обращения 24.12.2023г.).
2. Руководство по работе в программной среде *ROBOGUIDE V 6.40*, режим доступа: <http://belfingroup.com/belfingrupp/spravka/skachat/pdf-katalogi-funuc.html>. (дата обращения 24.12.2023г.).

## 8. УПРАВЛЕНИЕ ЦИФРОВЫМИ ВХОДАМИ/ВЫХОДАМИ В ПРОГРАММНОЙ СРЕДЕ ROBOGUIDE

Цель – приобретение навыков компьютерного проектирования роботизированного технологического комплекса (РТК) с промышленным роботом *FANUC* и периферийными динамическими объектами, управляемыми цифровыми входами/выходами в программной среде *ROBOGUIDE*.

С этой целью необходимо сформировать РТК в составе робота *FANUC*, оснащённого двухпальцевым захватным устройством, и двух подвижных конвейеров, управляемых цифровыми входами/выходами контроллера робота. Далее необходимо составить и отладить программу работы РТК в следующей последовательности. Контроллер робота через цифровой выход (*DO1*) активирует движение детали по первому конвейеру. Конечная позиция перемещения детали контролируется цифровым входом (*D11*). По сигналу этого входа активируется программа переноса детали роботом с первого конвейера на второй. Затем контроллер робота через цифровой выход (*DO2*) включает второй конвейер, который уносит деталь за пределы РТК.

Работа рассчитана на два часа.

### 8.1. КРАТКИЕ СВЕДЕНИЯ ИЗ ТЕОРИИ

Контроллер промышленного робота управляет не только роботом, но и периферийным оборудованием, входящим в состав РТК. В качестве такого оборудования могут использоваться конвейеры, питатели, позиционеры, металлорежущие станки и т.п. Управление осуществляется через цифровые выходы *DO* (*Digital Outputs*) и цифровые входы *DI* (*Digital Inputs*). Через выходы оборудование включается или выключается, а через входы контроллер получает информацию о состоянии оборудования (включено или выключено). В контроллерах *FANUC* предусмотрено 512 цифровых входов и 512 цифровых выходов. Каждый цифровой вход/выход имеет два устойчивых состояния – *ON* (включено или логическая единица) и *OFF* (выключено или логический ноль).

В программной среде *ROBOGUIDE* в процессе формирования проекта РТК необходимо предварительно настроить подвиж-

ное оборудование и панель управления цифровыми входами/выходами (*I/O*) контроллера робота. Для этого после открытия нового проекта добавляем подвижный конвейер из вкладки **Machine** → **Add Machine** → **CAD Library** дерева проекта (рис. 8.1).

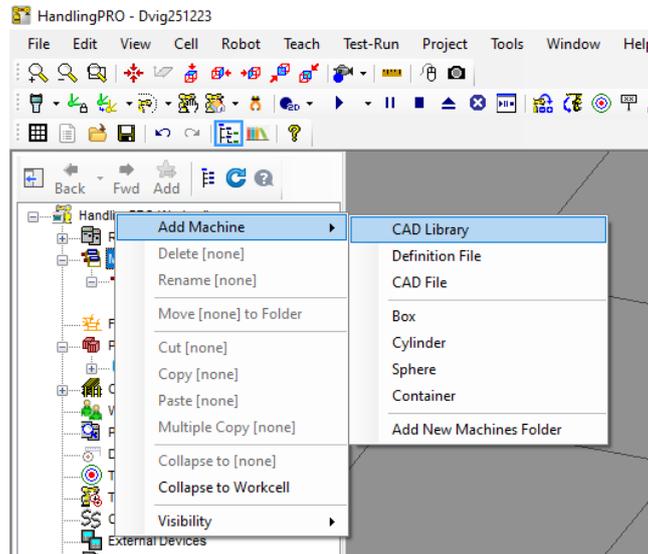


Рис. 8.1 Выбор подвижного оборудования

Из библиотеки CAD-моделей дерева проекта в разделе **conveyor** выбираем конвейер **snvug**. Ориентируем выбранный конвейер по отношению к роботу, как показано на рис. 8.2.

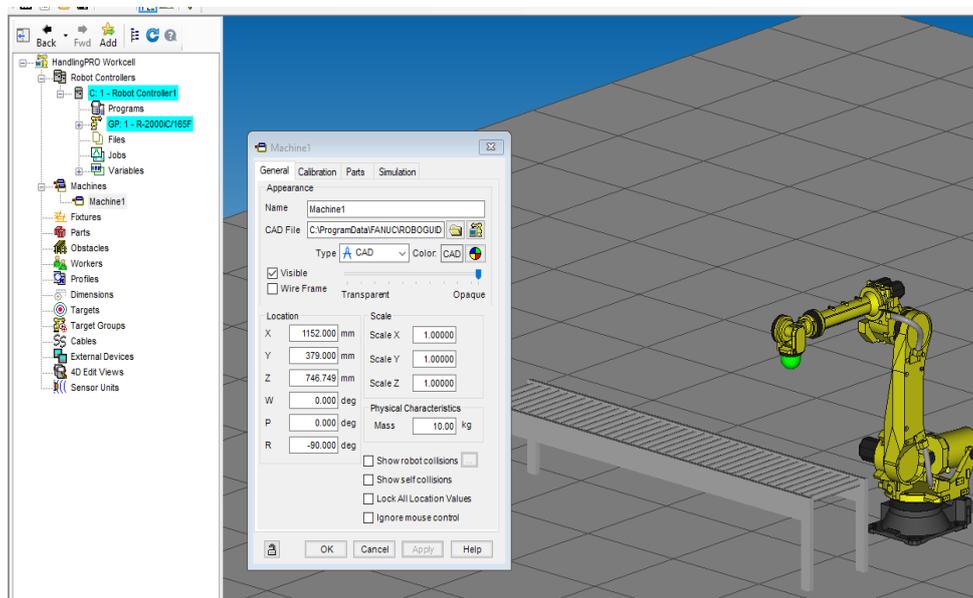


Рис. 8.2 Выбор конвейера

Далее «оснащаем» конвейер электрическим приводом. Для этого в дереве проекта активируем правой кнопкой мыши надпись **Machine1**, а в появившемся окне выбираем **Add Link**

(добавить подвижную «ссылку»). Подвижная ссылка в виде параллелепипеда (**Box**) добавится в проект (рис. 8.3).

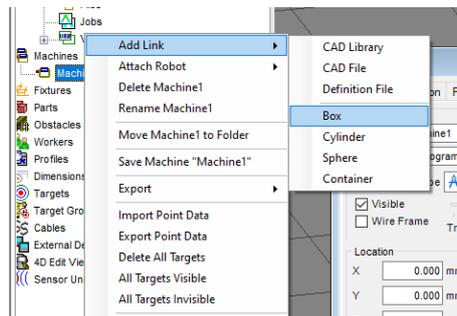


Рис. 8.3 Добавление подвижной «ссылки»

Подвижная «ссылка» – это виртуальный подвижный объект, к которому далее будет «привязана» подвижная деталь. Эту «ссылку» необходимо уменьшить в размерах во вкладке **Link CAD** окна свойств этого объекта (рис. 8.4).

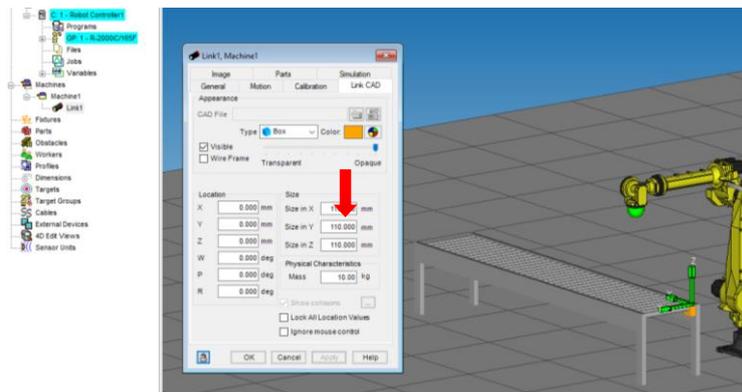


Рис. 8.4 Подвижная «ссылка»

Далее, перейдя во вкладку **General**, ставим галочку в строке **Edit Axis Origin**. У конвейера появится изображение электродвигателя со своей системой координат, которую необходимо сориентировать так, чтобы её ось **Z** совпадала с направлением движения детали на конвейере (рис. 8.5).

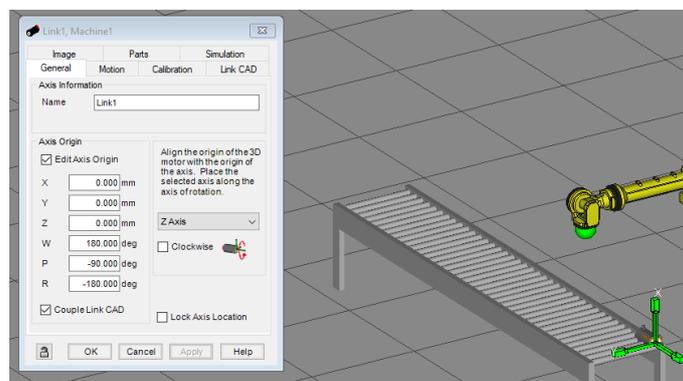


Рис. 8.5 Ориентация двигателя

Чтобы внести в проект деталь, необходимо в дереве проекта активировать правой кнопкой мыши строку **Parts** и далее **Add Part** → **Box**. Деталь в виде куба появится на подложке в стороне от РТК. Рекомендуется изменить её цвет с зелёного на какой-либо другой, например, розовый. Чтобы «привязать» деталь к конвейеру, необходимо в дереве проекта дважды щёлкнуть левой кнопкой мыши на строке **Link1** и во вкладке **Parts** окна свойств этого объекта поставить галочку в квадрате **Part1** и нажать клавишу **Apply** (применить). Чтобы выполнить ориентацию детали на конвейере в позиции её взятия роботом (рис. 8.6) необходимо в той же вкладке поставить галочку в квадрате **Edit Part Offset**. Сориентировать систему координат детали необходимо так, как показано на рис. 8.6.

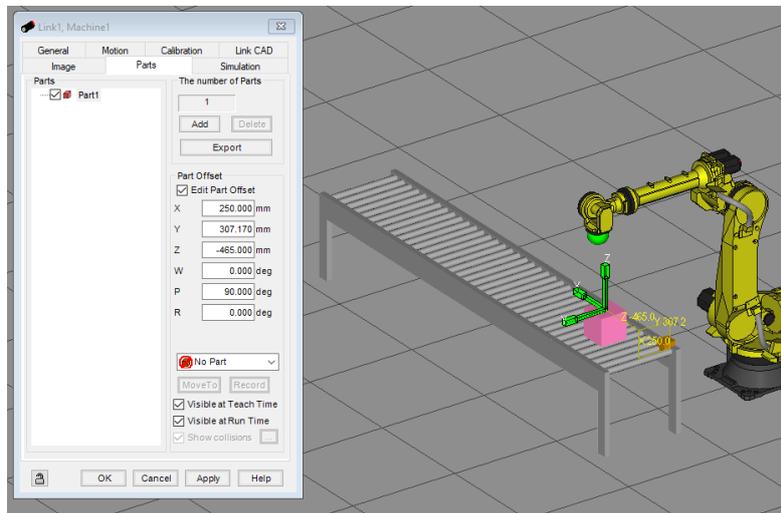
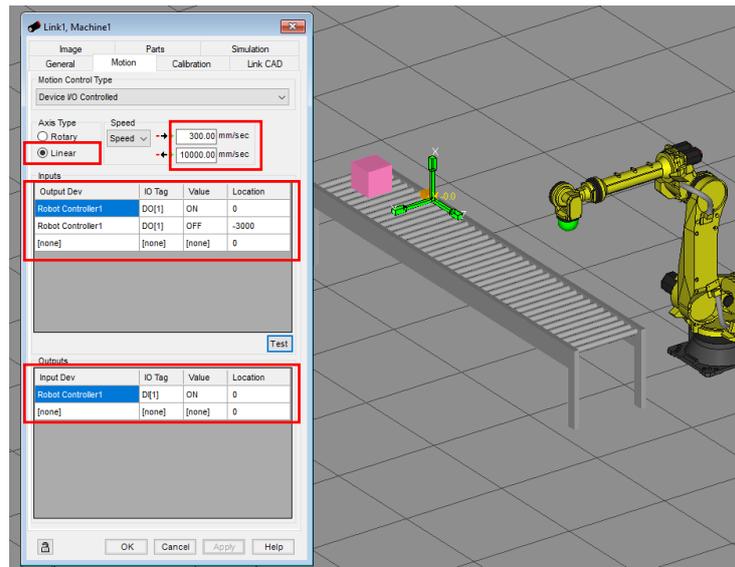


Рис. 8.6 Ориентация детали на подвижном конвейере

Чтобы заставить деталь двигаться по конвейеру, необходимо корректно ввести параметры во вкладке **Motion** (движение). В строке **Motion Control Type** этой вкладки выбрать **Device I/O Controlled** (управляемое устройство ввода/вывода). После этого появится обновлённая вкладка **Motion**. В этой вкладке необходимо ввести (изменить) параметры, как показано на рис. 8.7. Чтобы выбрать параметры ячеек таблиц **Inputs** и **Outputs**, достаточно подвести курсор в нужную ячейку и нажать левую кнопку мыши. После введения всех параметров надо нажать клавишу **Apply** и проверить корректность движения детали по конвейеру нажатием кнопки **Test** во вкладке **Motion**.

Рис. 8.7 Параметры вкладки **Motion**

Отметим, что согласно заполненным таблицам в этой вкладке, перевод детали в начальную точку движения осуществляется цифровым выходом со значением *OFF* (**DO[1]=OFF**), а перевод в позицию её захвата роботом – цифровым выходом со значением *ON* (**DO[1]=ON**). Перевод детали в позицию захвата роботом сигнализирует цифровой вход со значением *ON* (**DI[1]=ON**).

Завершающим этапом формирования подвижного объекта является заполнение панели «входы/выходы». Эта панель вызывается из главного меню во вкладке **Tools** (рис. 8.8).

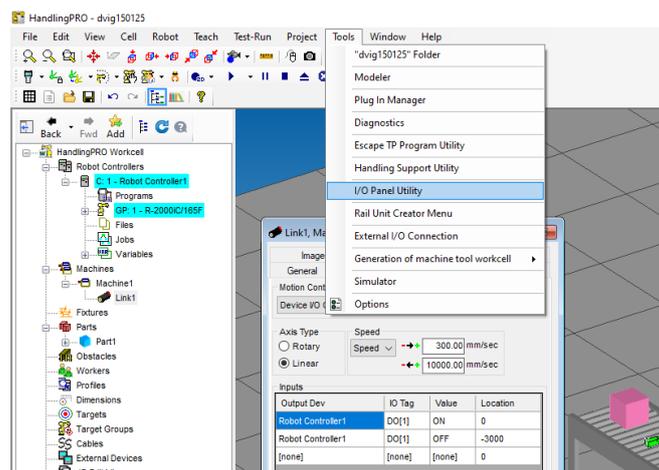


Рис. 8.8 Вызов панели «входы/выходы»

На этой панели необходимо вызвать таблицу **Monitoring Signals** нажав ссылку по красной стрелке (рис. 8.9). Информация о входах/выходах выбирается в правом верхнем углу окна и заносится в таблицу нажатием клавиши **Add**.

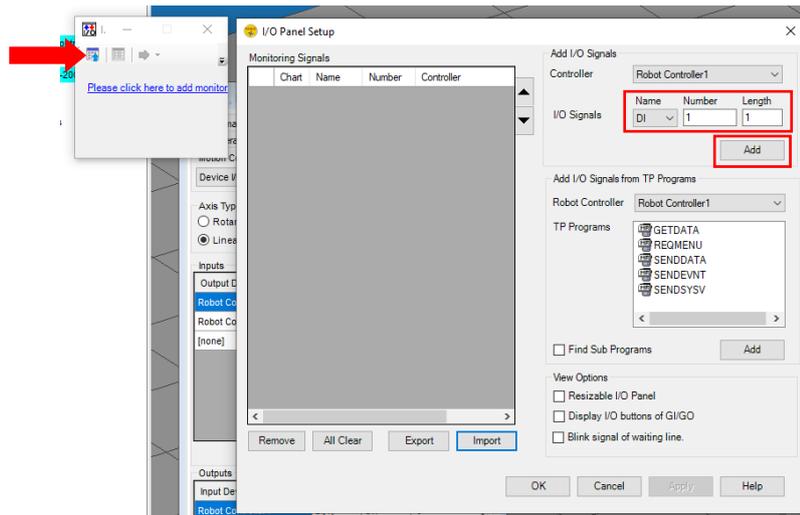


Рис. 8.9 Панель «входы/выходы»

Результат заполнения таблицы «входы/выходы» представлен на рис. 8.10. После нажатия клавиши **Apply** на панели появилась кнопка, помеченная красной стрелкой, нажатием которой можно управлять положением детали на конвейере.

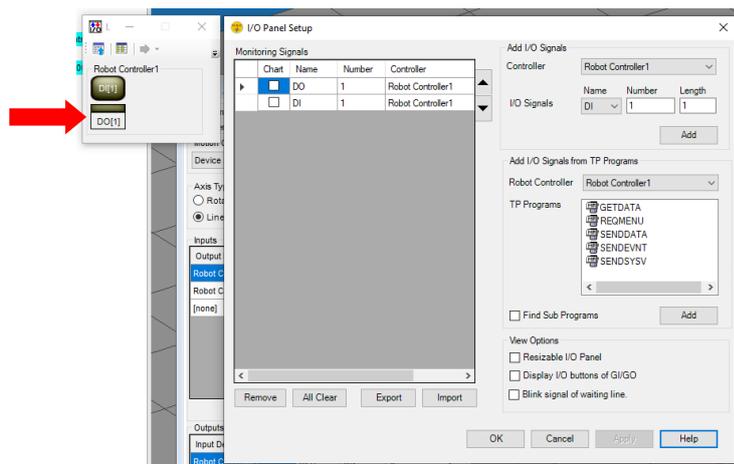


Рис. 8.10 Заполненная панель «входы/выходы»

Дальнейшее формирование проекта изучено на предыдущих практических занятиях. Поэтому здесь рассматривается кратко. Оно включает оснащение робота двухпальцевым захватом с привязкой к нему детали. Добавление в проект фиксирующего объекта в виде стола на ножках и привязка к нему детали (рис. 8.11).

После формирования РТК составляется программа работы РТК. Напомним, что для этого необходимо в дереве проекта

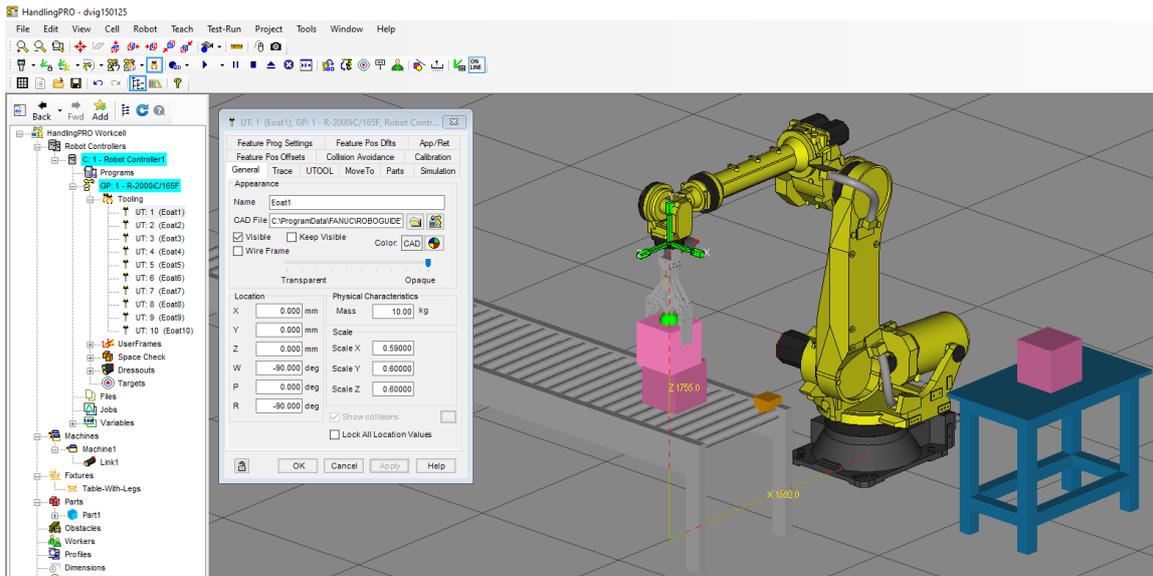


Рис. 8.11 Сформированный проект РТК

активировать правой кнопкой мыши надпись **Programs** и выбрать во всплывшем меню **Add Simulation Program**. Составленная программа рабочего цикла РТК включает активацию работы подвижного конвейера, при этом деталь начинает двигаться в рабочую позицию захвата роботом. По достижению позиции захвата робот берёт деталь, помещает её на рабочий стол и возвращается в домашнее положение. Программа рабочего цикла представлена на рис. 8.12.

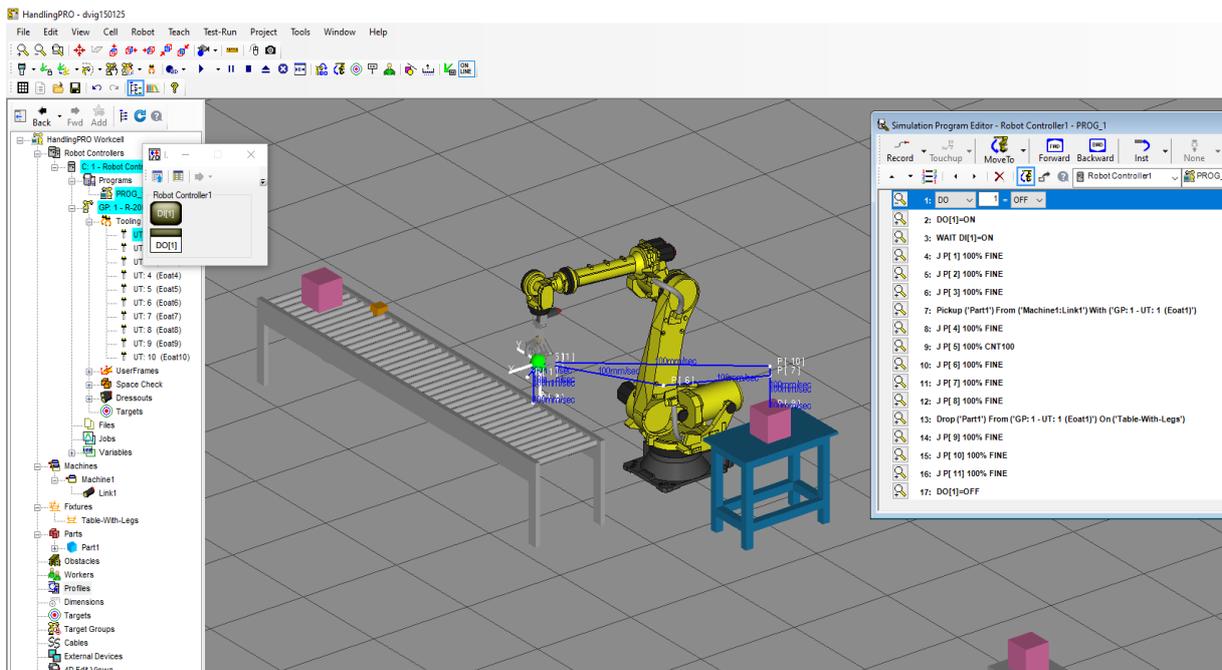


Рис. 8.12 Программа рабочего цикла РТК

Покажем, как в программе используются цифровые входы/выходы. Присвоением цифровому выходу **DO[1]** значения **OFF** в первой строке проверяется положение детали в позиции начала движения. Во второй строке **DO[1]=ON** запускается движение детали по конвейеру в позицию её захвата роботом. В третьей строке запрограммировано ожидание, пока деталь не придёт в позицию захвата и не появится на цифровом входе **DI[1]** значение **ON**. После этого робот начнёт свою работу (строки 4 ... 16). Последняя команда в строке 17 дублирует первую (строка 1).

## 8.2. ЗАДАНИЕ

В программной среде *ROBOGUIDE* сформировать проект РТК, состоящего из робота *FANUC M-710iC/12L*, оснащённого электрическим двухпальцевым захватом, и двух подвижных конвейеров. Захват импортировать из библиотеки *CAD-моделей ROBOGUIDE*. Конвейеры расположить перпендикулярно друг к другу. Привязать деталь в виде куба к захвату робота и подвижным конвейерам.

Составить программу работы РТК, включающую движение детали с одного конца первого конвейера в позицию её захвата роботом. Далее робот берёт эту деталь, перекладывает её на второй конвейер и возвращается в домашнюю позицию. В завершении рабочего цикла контроллер робота включает второй конвейер, и деталь уносится за пределы РТК.

Продемонстрировать преподавателю работу РТК по составленной программе.

## 8.3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Каково назначение цифровых выходов *DO* (*Digital Outputs*) и цифровых входов *DI* (*Digital Inputs*)?
2. В какой вкладке дерева проекта формируется динамический объект периферийного оборудования?
3. Что такое подвижная «ссылка»?
4. Как ориентируется система координат электродвигателя подвижной «ссылки»?
5. Какой командой программируется ожидание появления сигнала на цифровом входе?

#### 8.4. ЛИТЕРАТУРА

1. Официальный сайт компании *FANUC*, режим доступа [www.fanuc.eu/ru/ru](http://www.fanuc.eu/ru/ru) . (дата обращения 24.12.2023г.).
2. Руководство по работе в программной среде *RO-BOGUIDE V 6.40*, режим доступа: <http://belfingroup.com/o-belfingrupp/spravka/skachat/pdf-katalogi-funuc.html>. (дата обращения 24.12.2023г.).