

# KUKA

Training

KUKA Roboter GmbH

## Применение и программирование промышленных роботов

Целевая группа Ученики и студенты



Состояние на: 20.01.2014

Версия: Edu Pack Einsatz und Programmierung von Industrierobotern V4



© Авторское право 2014

KUKA Roboter GmbH  
Zugspitzstraße 140  
D-86165 Augsburg  
Германия

Копирование данной документации, полностью или частично, равно как и предоставление ее третьим лицам разрешается только с однозначного разрешения компании KUKA Roboter GmbH.

В системе управления могут работать другие, не описанные в данной документации функции. Несмотря на это, при новой поставке или при сервисном обслуживании право на данные функции отсутствует.

Мы проверили содержание печатного документа на соответствие описанному аппаратному и программному обеспечению. Однако, так как отклонения исключить невозможно, полное соответствие нами не гарантируется. Содержащиеся в данном печатном документе сведения регулярно проверяются, и необходимые исправления включаются в следующее издание.

Возможны технические изменения, не оказывающие влияния на работу.

Перевод оригинальной документации

KIM-PS5-DOC

Публикация:	Pub COLLEGE Einsatz und Programmierung von Industrierobotern (TG-COL) ru
Структура книги:	EduPack Einsatz und Programmierung von Industrierobotern V5.1
Версия:	Edu Pack Einsatz und Programmierung von Industrierobotern V4

## Содержание

<b>1</b>	<b>Введение в робототехнику</b>	<b>7</b>
1.1	Обзор	7
1.2	Введение	7
1.3	R.U.R. – Россумские универсальные роботы	8
1.4	Законы робототехники	8
1.5	Первый робот	9
1.6	История компании KUKA	10
<b>2</b>	<b>Области применения промышленных роботов</b>	<b>13</b>
2.1	Обзор	13
2.2	Применение промышленных роботов	13
2.3	Примеры применения роботов	16
<b>3</b>	<b>Обзор компонентов робототехнической системы</b>	<b>27</b>
3.1	Обзор	27
3.2	Компоненты робототехнической ячейки	27
3.3	Выбор робота	28
3.4	Конфигурация системы управления	30
3.5	Выбор исполнительного органа/инструмента	30
3.6	Выбор системы энергоснабжения	31
3.7	Подсоединение периферийных устройств (с помощью магистральной шины) ..	32
3.8	Использование датчиков	33
3.9	Предохранительные устройства	33
<b>4</b>	<b>Промышленный робот</b>	<b>39</b>
4.1	Обзор	39
4.2	Запуск робота	39
4.3	Описание и конструкция	40
4.4	Механика робота KUKA	42
4.5	Расположение главных осей	44
4.6	Абсолютная точность и стабильность повторяемости	47
<b>5</b>	<b>Система управления роботом</b>	<b>51</b>
5.1	Обзор	51
5.2	Габаритные размеры системы управления роботом	53
5.3	Минимальные расстояния для системы управления роботом	54
5.4	Обзор системы управления роботом	55
5.5	Обзор приложений и шинных систем	59
5.5.1	Шина контроллера KUKA, KCB	61
5.5.2	Системная шина KUKA, KSB	62
5.5.3	Шина расширений KUKA Extension Bus, KEB	63
5.5.4	Линейный интерфейс KUKA Line Interface, KLI	64
5.6	Эффективность использования энергии	65
<b>6</b>	<b>Перемещение робота</b>	<b>67</b>
6.1	Обзор	67
6.2	Переносное программирующее устройство KUKA smartPAD	67
6.2.1	Передняя сторона	67

6.2.2	Обратная сторона .....	69
6.3	Чтение и интерпретация сообщений системы управления роботом .....	70
6.4	Выбор и установка режима работы .....	72
6.5	Перемещение осей робота по отдельности .....	74
6.6	Системы координат относительно робота .....	79
6.7	Перемещение робота в универсальной системе координат .....	81
6.8	Перемещение робота в системе координат инструмента .....	86
6.9	Перемещение робота в основной системе координат .....	89
<b>7</b>	<b>Ввод в эксплуатацию .....</b>	<b>95</b>
7.1	Обзор .....	95
7.2	Принцип юстировки .....	95
7.3	Юстировка робота .....	98
7.4	Нагрузки на роботе .....	101
7.5	Данные нагрузки инструмента .....	102
7.6	Дополнительные нагрузки на робот .....	106
7.7	Калибровка инструмента .....	108
7.8	Калибровка базы .....	116
7.9	Запрос текущего положения робота .....	121
<b>8</b>	<b>Выполнение программы робота .....</b>	<b>125</b>
8.1	Обзор .....	125
8.2	Выполнение перемещения инициализации .....	125
8.3	Выбор и запуск программ робота .....	126
<b>9</b>	<b>Обращение с файлами программы .....</b>	<b>133</b>
9.1	Обзор .....	133
9.2	Создание программных модулей .....	133
9.3	Обработка программных модулей .....	135
<b>10</b>	<b>Создание и изменение запрограммированных перемещений .....</b>	<b>137</b>
10.1	Обзор .....	137
10.2	Создание новой команды перемещения .....	137
10.3	Создание перемещений с оптимизацией времени такта (осевое перемещение) .....	139
10.4	Создание перемещений по траекториям .....	145
10.5	Изменение команд перемещения .....	155
<b>11</b>	<b>Использование логических функций в программе робота .....</b>	<b>161</b>
11.1	Обзор .....	161
11.2	Введение в программирование логики .....	161
11.3	Программирование функций ожидания .....	162
11.4	Программирование простых функций переключения .....	167
11.5	Программирование функций переключения траектории .....	170
<b>12</b>	<b>Введение в уровень эксперта .....</b>	<b>175</b>
12.1	Обзор .....	175
12.2	Использование уровня эксперта .....	175
<b>13</b>	<b>Циклы, обусловленные команды и различение ситуаций .....</b>	<b>179</b>
13.1	Обзор .....	179



13.2	Контроль выполнения программы .....	179
13.3	Циклы .....	180
13.4	Обусловленные команды и различие ситуаций .....	183
<b>14</b>	<b>Подпрограммы и функции .....</b>	<b>185</b>
14.1	Обзор .....	185
14.2	Работа с локальными подпрограммами .....	185
14.3	Работа с глобальными подпрограммами .....	187
14.4	Передача параметров в подпрограммы .....	189
<b>15</b>	<b>Переменные и описания .....</b>	<b>191</b>
15.1	Обзор .....	191
15.2	Хранение данных в KRL .....	191
15.3	Работа с простыми типами данных .....	194
15.3.1	Объявление переменных .....	194
15.3.2	Инициализация переменных с простыми типами данных .....	197
15.3.3	Манипуляция значениями переменных простых типов данных с помощью KRL .....	199
15.4	Массивы/поля с KRL .....	202
15.5	Структуры с KRL .....	205
15.6	Перечислимый тип данных ENUM .....	208
<b>16</b>	<b>Программирование перемещений с помощью KRL .....</b>	<b>211</b>
16.1	Обзор .....	211
16.2	Программирование перемещений с помощью KRL .....	211
16.3	Расчет или манипуляция позициями робота .....	220
16.4	Целенаправленное изменение битов состояния и поворота .....	221
<b>17</b>	<b>Работа с системой управления верхнего уровня .....</b>	<b>227</b>
17.1	Обзор .....	227
17.2	Подготовка к запуску программы от ПЛК .....	227
17.3	Настройка соединения с ПЛК (Cell.src) .....	229
17.4	Конфигурирование и применение режима «Внешняя автоматика» .....	231
<b>18</b>	<b>Программирование с помощью WorkVisual .....</b>	<b>243</b>
18.1	Обзор .....	243
18.2	Соединение с системой WorkVisual .....	243
18.3	Управление проектом с помощью WorkVisual .....	251
18.3.1	Открытие проекта с помощью WorkVisual .....	251
18.3.2	Сравнение проектов с помощью WorkVisual .....	255
18.3.3	Передача проекта в систему управления роботом (установка) .....	259
18.3.4	Активация проекта в системе управления роботом .....	263
18.4	Редактирование программ KRL с помощью WorkVisual .....	266
18.4.1	Работа с файлами .....	266
18.4.2	Работа с редактором KRL .....	272
<b>19</b>	<b>Приложение .....</b>	<b>279</b>
19.1	Сокращения .....	279
19.2	Используемые термины .....	280
19.3	Выдержки из описания «Безопасность KR C4» .....	282
	<b>Индекс .....</b>	<b>289</b>



# 1 Введение в робототехнику

## 1.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Введение
- R.U.R. – Россумские универсальные роботы
- Законы робототехники
- Первый робот
- История компании KUKA



Рис. 1-1: Смена главы

## 1.2 Введение



Рис. 1-2

Растущее разнообразие изделий и их вариантов в ассортименте для поддержания или повышения конкурентоспособности производственного предприятия требует, в том числе, повышения продуктивности и гибкости производства. Для необходимой для этого гибкой системы автоматизации оптимальным средством является использование промышленных роботов (IR).

Понятие «робот» происходит от славянского слова «robota», обозначающего тяжелую работу.

В технической терминологии термин «промышленный робот» обособлен от других средств автоматизации и технологических машин. Тем не менее, на международном уровне существует определенная путаница с терминами, так как к роботам нередко причисляются схожие системы,

например, манипуляторы и загрузочные устройства и учитывают их в статистических данных.

Это связано с тем, что конструкция всех данных механических систем представляет собой кинематическую цепочку с неподвижной частью и манипулятором (или несколькими манипуляторами), на который монтируется рука с захватом или инструментом (напр., сварочной горелкой).

### 1.3 R.U.R. – Россумские универсальные роботы

R.U.R. (с чешского: **Rosumovi Umeli Roboti**) – название изданной в 1921 г. драмы чешского писателя Карела Чапека.

В драме рассказывается о фирме, которая производит похожие на человека машины (роботы), которые облегчают людям жизнь. В ходе повествования эти машины начинают революцию в обществе и уничтожают человечество.

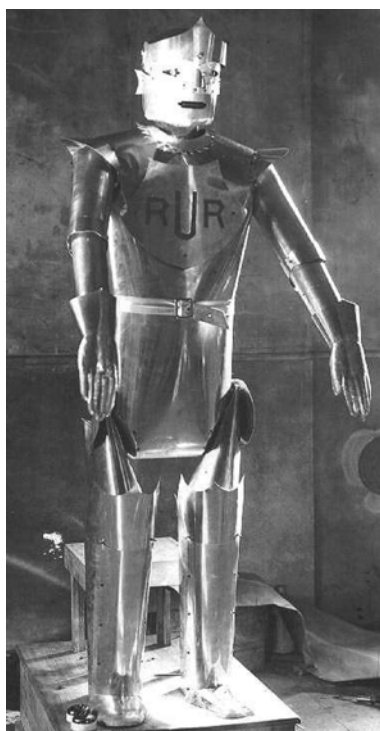


Рис. 1-3: R.U.R. – Россумские универсальные роботы

Название пьесы «R.U.R» расшифровывается как «Россумские универсальные роботы», по названию фирмы, которая производила эти машины. Фамилия Россум является ироничным намеком автора, так как чешское слово «rozum» (произносится как «росумм» с кратким первым слогом) обозначает благоразумие, здравый смысл. То есть дословный перевод оригинального названия звучит следующим образом: «Искусственные рабочие (господина) Разума», слова «Россум» и «универсальные» были использованы в переводе только для того, чтобы можно было сохранилось чешское сокращение «R.U.R.».

Слово «робот» из этой пьесы быстро стало использоваться в обыденной речи во многих странах мира. [8]

### 1.4 Законы робототехники

Законы робототехники были сформулированы еще **Айзеком Азимовым** в его сборнике фантастических рассказов «**I, Robot**» (1950 г.) и с тех пор легли в основу всеобщего представления о том, каким должен быть ро-

бот. Описанные Азимовым роботы в своих действиях и при принятии решений руководствуются данными законами робототехники.

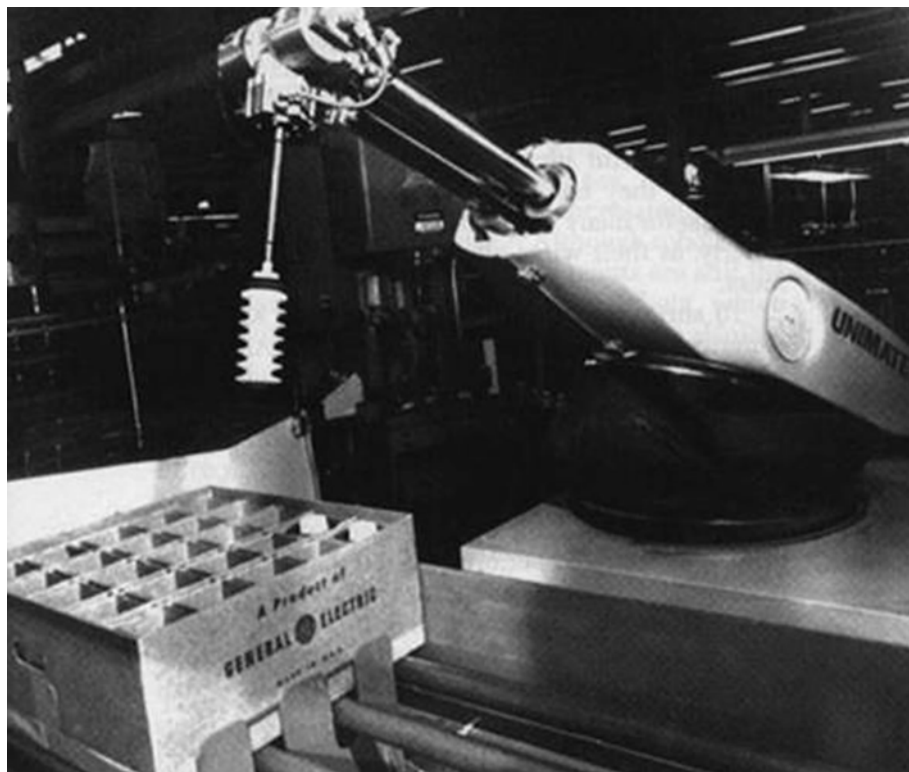
Эти законы действительны, прежде всего, для описанных в литературе роботов, но тем не менее они также повлияли на принципы программирования современных роботов и в измененной форме используются, например, в качестве лозунга на соревнованиях роботов-уборщиков. Все современные промышленные роботы тоже программируются в соответствии с законами Айзека Азимова, хотя программисты часто не задумываются об этом.

Законы робототехники сформулированы следующим образом:

1. Робот не может причинить вред человеку или своим бездействием допустить, чтобы человеку был причинен вред.
2. Робот должен повиноваться командам человека, если эти команды не противоречат Первому Закону.
3. Робот должен заботиться о своей безопасности, если это не противоречит Первому и Второму законам.

Следует учитывать, что законы имеют определенную иерархию. Даже если эти законы кажутся ясными, они не могут полностью защитить от ошибок, особенно потому, что люди рассматривают их с человеческой точки зрения и, следовательно, считают их несовершенными. [8]

## 1.5 Первый робот



**Рис. 1-4: Первый промышленный робот (тип: Unimate, изготовитель: Unimation, ввод в эксплуатацию: 1961 г.)**

Первый промышленный робот, который позднее был назван «Unimate», появился в результате обсуждения фантастического романа при встрече изобретателей Джорджа Девола и Джозефа Энгельбергера, которая произошла в 1956 году. Вдохновившись романом, изобретатели решили разработать первого реального робота.

В 1962 г. робот «Unimate» был интегрирован в конвейерное производство компании General Motors (Трентон, США). Задача этого робота состояла в том, чтобы принимать горячие заготовки из прессы для металла и складывать их в стопки. Программа робота состояла из множества отдельных шаговых команд, которые были сохранены в памяти на магнитном барабане. Благодаря этому он уже мог выполнять различные автоматические операции.

**1.6 История компании KUKA**

**1898 г.**

В 1898 г. предприниматели Йоханн Йозеф Келлер и Якоб Кнаппих основали Аугсбургский ацетиленовый завод.




**Рис. 1-5: Йоханн Йозеф Келлер (слева) и Якоб Кнаппих (справа)**

**История**

	<p><b>1900 г.</b> Ацетиленовый прожектор Кнаппиха</p>
	<p><b>1927 г.</b> Крупногабаритный мусоровоз для утилизации мусора городскими службами</p>
	<p><b>1939 г.</b> Станок для точечной сварки «Mars»</p>



	<p><b>1949 г.</b> Печатная машинка «Princess»</p>
	<p><b>1970 г.</b> Башня бронетранспортера «Marder»</p>

## Роботы KUKA

	<p><b>1973 г.</b> KUKA разрабатывает первый в мире промышленный робот с шестью осями, приводимыми в действие электродвигателями, который называется FAMULUS.</p>
	<p><b>1976 г.</b> Разработка нового типа робота с осями с электромеханическим приводом и угловым манипулятором.</p>
	<p><b>1989 г.</b> Новое поколение продукции с бесщеточными приводными двигателями и двойным угловым манипулятором для нагрузок от 8 кг до 240 кг, без кинематики в виде параллелограмма.</p>
	<p><b>1996 г.</b> Первая система управления роботом на базе ПК и новая линия роботов способствуют успеху компании KUKA Roboter GmbH. В последующие годы разрабатывается целый ряд новаторских программных приложений, например, RoboTeam и SafeRobot.</p>
	<p><b>2007 г.</b> KR 1000 titan стал самым сильным промышленным роботом в мире. Вариант этого робота, предназначенный для палетоукладки, может перемещать до 1300 кг груза.</p>

	<p><b>2011 г.</b></p> <p>Благодаря такой продукции, как KR C4, QUANTEC, smartPAD и KUKA автоматизация значительно упрощается.</p>
	<p><b>2012 г.</b></p> <p>Серия KR AGILUS с системой управления KR C4 comp – будущее малой робототехники.</p>

**Компания KUKA сегодня**

Компания KUKA Roboter GmbH с главным офисом в Аугсбурге является одним из предприятий акционерного общества KUKA и во всем мире считается ведущим производителем промышленных роботов. Главной задачей данной компании является разработка и производство, а также сбыт промышленных роботов, систем управления, программного обеспечения и линейных блоков. Предприятие является лидером на рынках Германии и Европы и занимает второе место в мире. В KUKA Robotics во всем мире работает более 3 180 человек. Отдел производства робототехники в 2012 г. обеспечил доход от реализации в размере 742,6 млн евро. 25 дочерних предприятий компании представляют ее на важнейших рынках Европы, Америки и Азии.



**Рис. 1-6: Предприятия KUKA в Аугсбурге и Герстхофене**

## 2 Области применения промышленных роботов

### 2.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Статистические данные о робототехнических системах
- Примеры применения



Рис. 2-1: Смена главы

### 2.2 Применение промышленных роботов

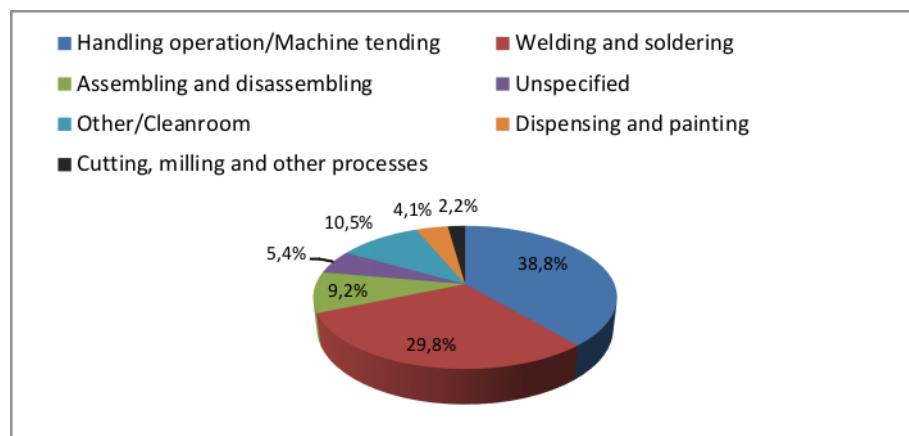


Рис. 2-2: Области применения роботов

**Легенда:**

[Источник: IFR World Robotics 2011]

#### Смена инструмента

К роботам для смены инструмента относятся:

- Роботы для точечной сварки
- Роботы для контурной сварки
- Роботы для напыления
- Роботы для зачистки заусенцев

#### Манипулирование заготовками

К роботам для манипулирования заготовками относятся:

- Роботы для манипулирования заготовками под стационарными инструментами (стационарными клещами для точечной сварки, клещами для клепки, горелкой)
- Роботы для загрузки металлообрабатывающих станков и обрабатывающих центров

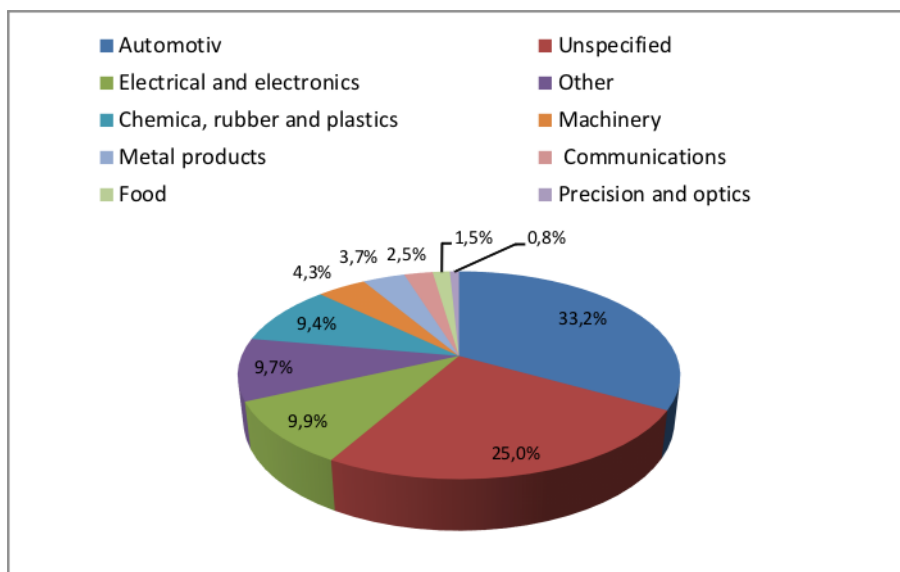
- Роботы для загрузки и разгрузки прессов и машин для литья под давлением
- Роботы для манипулирования заготовками на ковочных, отжигательных и закалочных установках, а также на установках по производству стеклянных изделий

**Монтаж**

Монтаж:

- Роботы, применяемые в автомобилестроении для монтажа колес, раздвижных крыш и оконных стекол
- Роботы, применяемые для монтажа экранных устройств изображения

**Роботы в промышленности**



**Рис. 2-3: Отрасли промышленности**

**Легенда:**

[Источник: IFR World Robotics 2011]

**Концентрация роботов 2010**

Количество роботов в производстве на 10 000 сотрудников, [источник: IFR World Robotics 2011]

**Страны с концентрацией роботов выше среднего (51)**

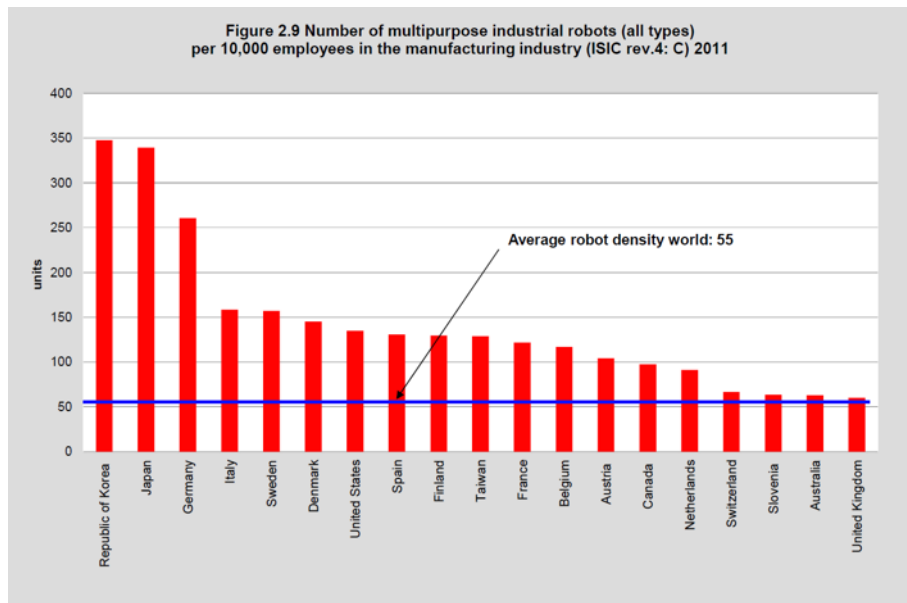


Рис. 2-4: Концентрация роботов 1

Легенда:

Источник: IFR World Robotics 2012

Страны с концентрацией роботов ниже среднего (51)

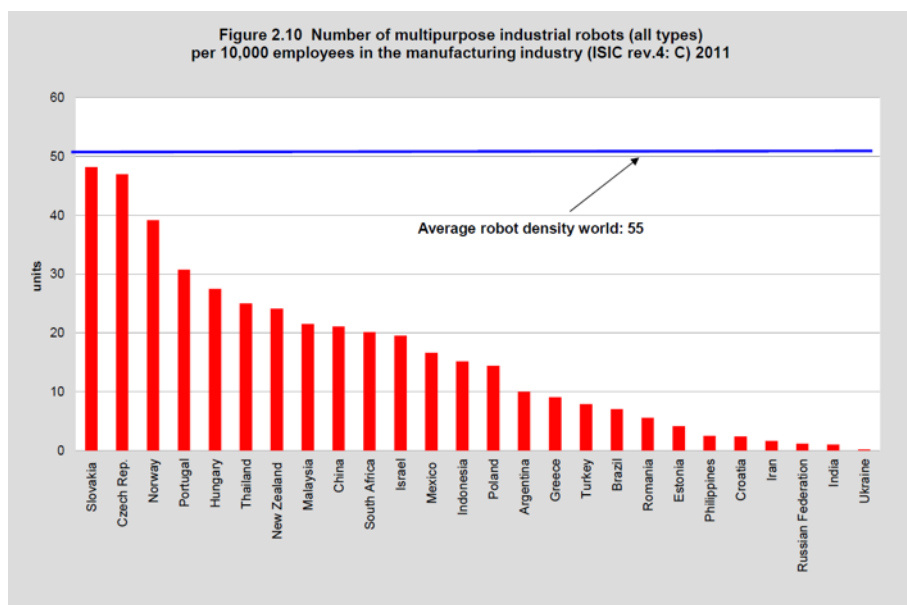


Рис. 2-5: Концентрация роботов 2

Легенда:

Источник: IFR World Robotics 2012

1

Концентрация роботов в автомобилестроении к различным странам

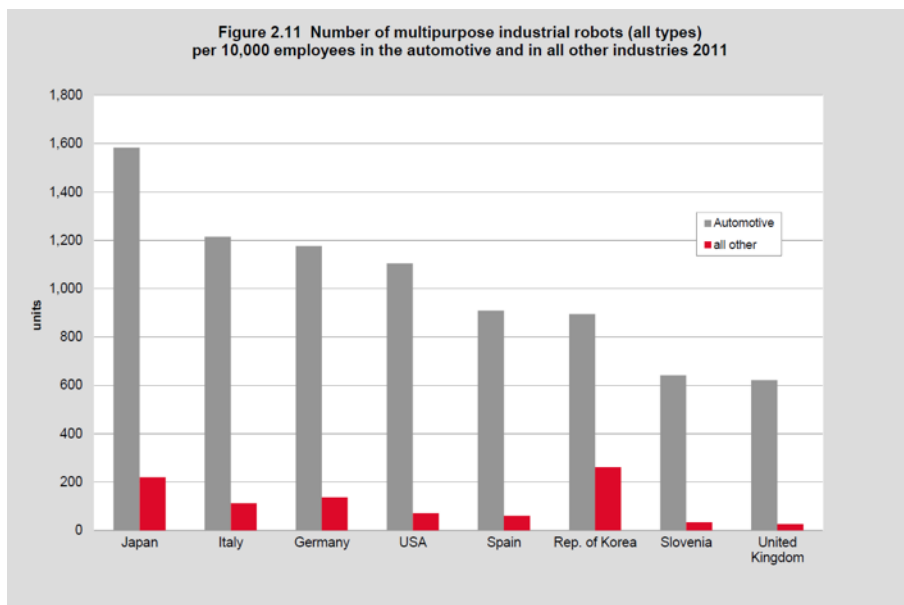


Рис. 2-6: Концентрация роботов в автомобильной промышленности

Легенда:

Источник: IFR World Robotics 2012

### 2.3 Примеры применения роботов

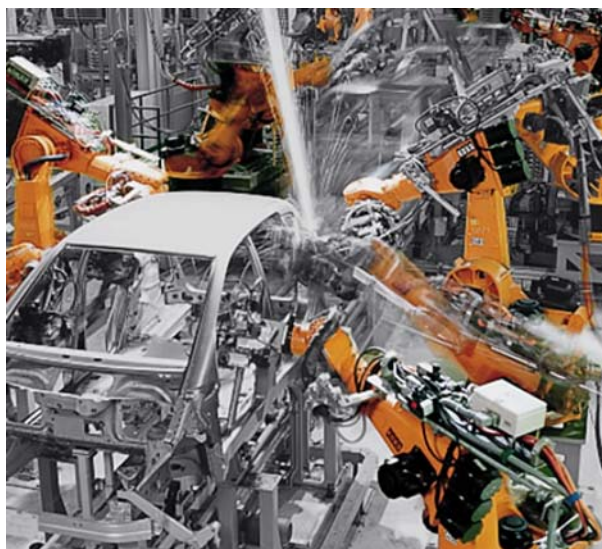


Рис. 2-7: Точечная сварка неотделанного кузова





Рис. 2-8: Сварка в среде защитного газа

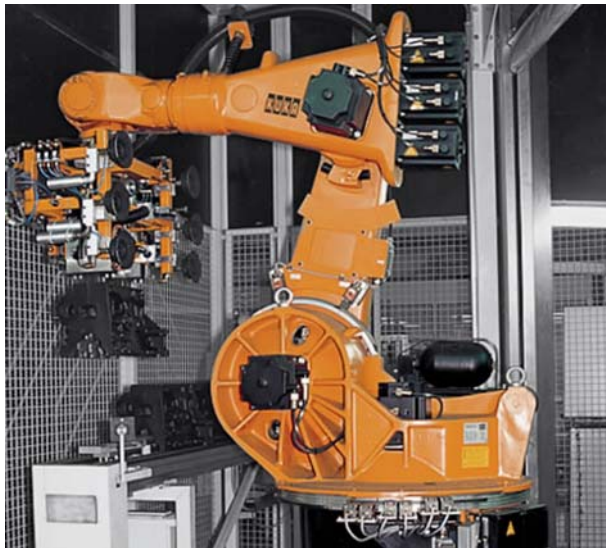


Рис. 2-9: Манипуляции с блоками цилиндров двигателей



Рис. 2-10: Манипулирование пивными бочками

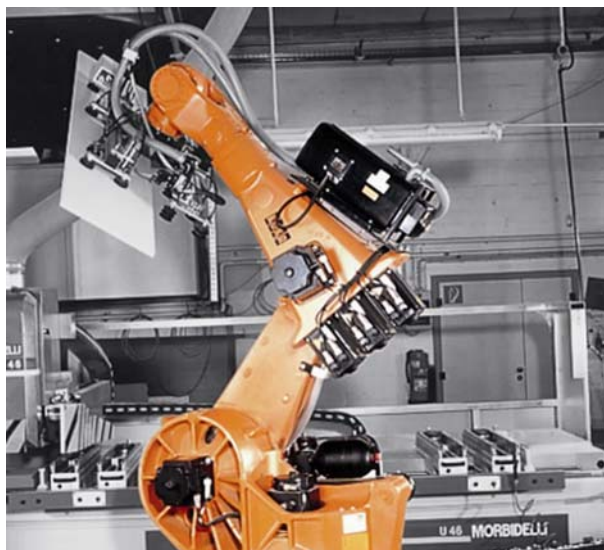


Рис. 2-11: Манипулирование деталями мебели



Рис. 2-12: Манипулирование листовыми деталями – гибка листов



Рис. 2-13: Манипулирование листовыми деталями – межпрессовые системы



Рис. 2-14: Обработка – шлифование и полировка





Рис. 2-15: Обработка – плазменная резка литых деталей

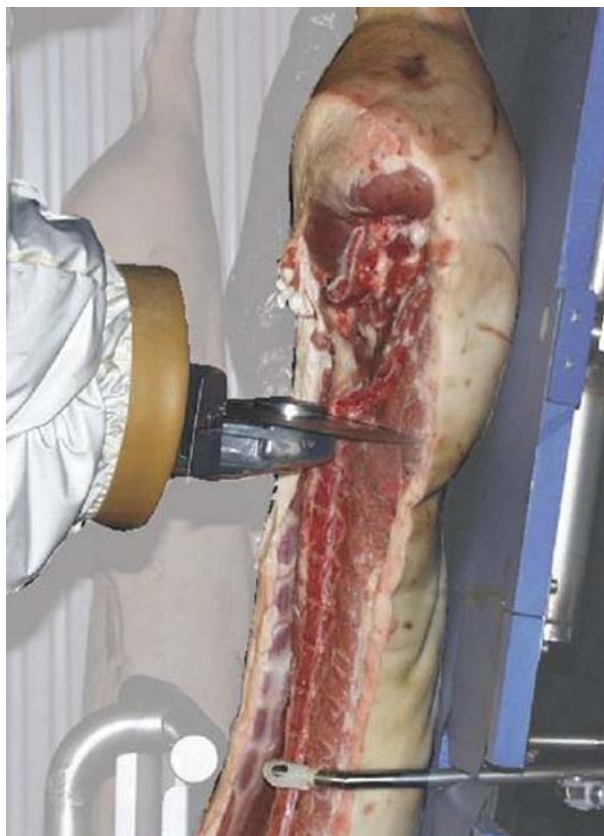


Рис. 2-16: Обработка – разделка мяса



**Рис. 2-17: Обработка пищевых продуктов**



**Рис. 2-18: Монтаж – стыковка резиновых прокладок**



Рис. 2-19: Монтаж автомобильных сидений



Рис. 2-20: Палетирование картонных коробок





Рис. 2-21: Палетирование мешков с химикатами



Рис. 2-22: Укладка на палеты ящиков для напитков

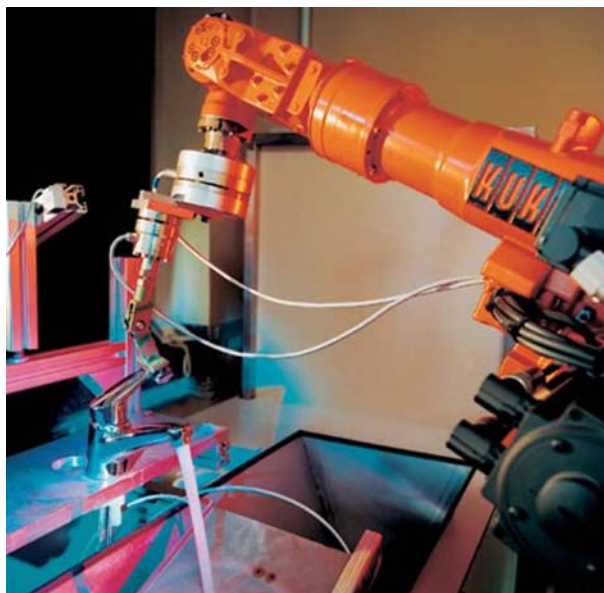


Рис. 2-23: Испытание водопроводной арматуры



Рис. 2-24: Испытания – испытание автомобильных сидений на долговечность



**Рис. 2-25: Измерение – встроенные лазерные измерительные устройства**



**Рис. 2-26: Измерения в аэродинамической трубе – детальное изображение**

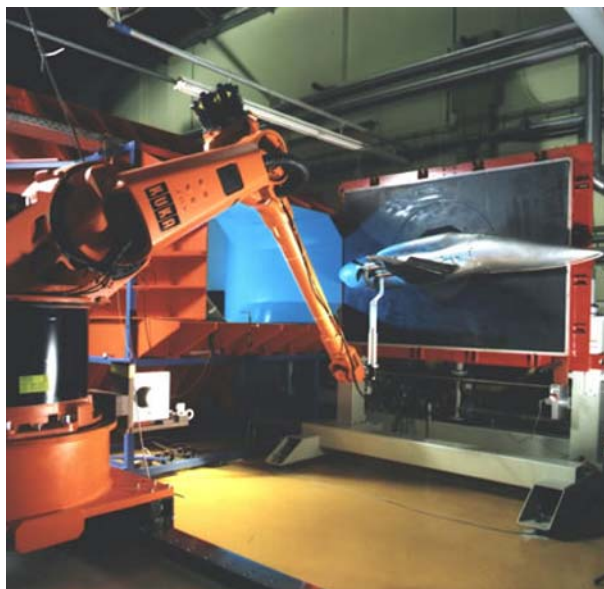


Рис. 2-27: Измерения в аэродинамической трубе



Рис. 2-28: Индустрия развлечений – ROBOCOASTER



Рис. 2-29: Медицинское оборудование/устройства

## 3 Обзор компонентов робототехнической системы

### 3.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Компоненты робототехнической ячейки
- Критерии выбора робота
- Управление осями робота и дополнительными осями
- Выбор инструмента
- Выбор системы энергоснабжения
- Подсоединение периферийных устройств
- Использование датчиков
- Предохранительные устройства



Рис. 3-1: Смена главы

### 3.2 Компоненты робототехнической ячейки

Как правило, робототехническая система/ячейка состоит из следующих компонентов:

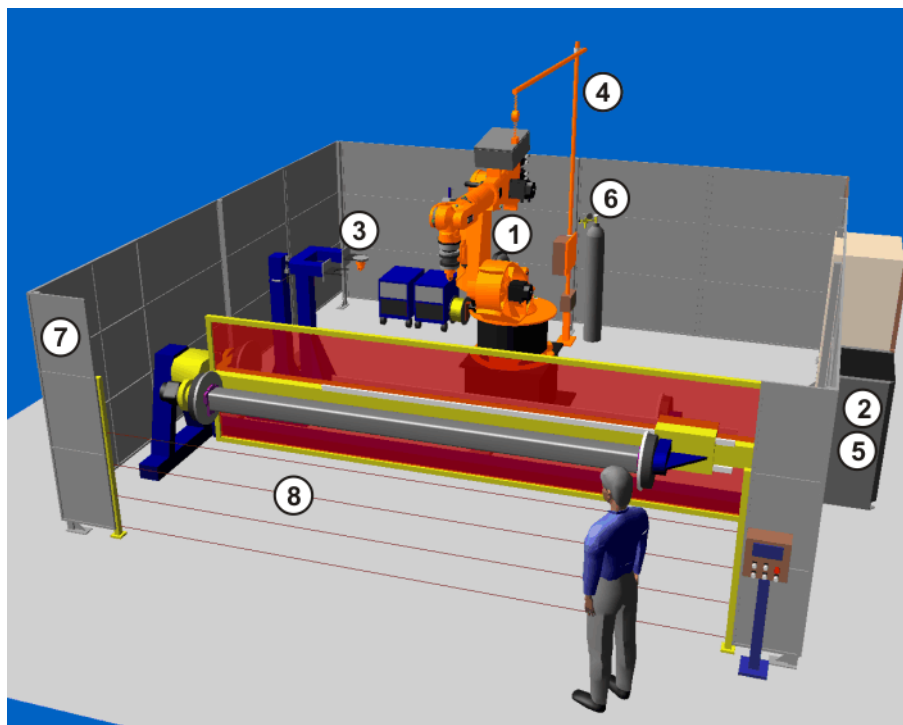


Рис. 3-2: Сварная секция для сварки в среде защитного газа



Поз.	Описание
1	Робот
2	Системы управления
3	Инструмент/устройство смены инструментов
4	Система подачи энергии
5	Система подсоединения периферийных устройств
6	Датчики
7	Защитное ограждение
8	Зона загрузки со световой завесой

### 3.3 Выбор робота

#### Критерии для выбора робота:

##### Нагрузки

- **Нагрузка:** нагрузка – это совокупность массы, момента инерции, статических и динамических сил, которые воздействуют на робота.
- **Номинальная нагрузка:** максимальная нагрузка, которая может воздействовать на фланец робота в нормальных условиях (температура, влажность воздуха ...) и без снижения какой-либо характеристики нагрузки.
- **Дополнительная нагрузка:** нагрузка, которую робот может взять на себя дополнительно к номинальной нагрузке. Дополнительная нагрузка устанавливается на манипулятор робота, балансир и/или карусель.

##### Условия эксплуатации

- **Применение:** изготовитель определяет главный тип(-ы) эксплуатации, предусмотренный для робота.

Примеры типичных случаев применения:

- Манипулирование
- Монтаж
- Точечная сварка
- Контурная сварка
- Нанесение клея/уплотнительного средства
- Обработка материалов (фрезерование)
- **Нормальные условия (окружающей среды):** (EN ISO 9946) производитель указывает предельные значения окружающей среды, при которых можно достигнуть заданный уровень производительности. Эти значения должны предоставляться для правильного хранения и эксплуатации, если имеются отличия. Условия окружающей среды охватывают следующие параметры, но не обязательно ограничиваются ими:
  - Температура (°C)
  - Относительная влажность воздуха (%)
  - Предельная высота расположения (м)
  - Электромагнитные помехи
- Стабильность повторяемости и абсолютная точность

##### Характеристики пространства

- **Рабочее пространство:** часть ограниченного пространства, которое фактически используется после выключения всех задаваемых прикладной программой движений. Размер пространства ограничивается точкой запястья.



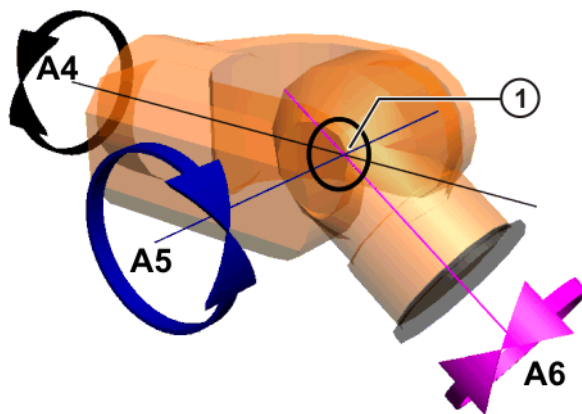


Рис. 3-3: Точка запястья

Рабочее пространство различается в зависимости от типа робота:

- Стандартные роботы
- Потолочные роботы
- Консольные роботы
- Роботы-палетоукладчики
- Роботы SCARA
- Роботы с линейной осью
- **Внешние габариты и масса:** для всей конструкции важны данные о габаритных размерах (мм) и весе робота (например, для транспортировки/установки/замены).
- **Базовая монтажная площадь:** описание базовой монтажной площади, необходимой для монтажа станины робота, чтобы обеспечить безопасную и надежную эксплуатацию.
- **Механический интерфейс:** чертеж с размерами согласно стандарту (ISO 9409-1).

### Скорость робота

для достижения продолжительности такта необходимо наличие значений максимальной скорости отдельных осей, а также максимальной номинальной скорости движения по траектории.

Определение продолжительности такта

- **Приблизительная оценка:** возможность неточной, но быстрой оценки продолжительности такта с помощью схемы или чертежа CAD. Результат также зависит от эмпирических значений.
- **Моделирование:** планирование робототехнической ячейки и моделирование процесса с помощью программы KUKA.Sim или другой программы для моделирования роботов (Robcad (em-Workplace)). Данный метод позволяет получить очень точные результаты.
- **Экспериментальная модель:** дорогостоящая возможность (материалы, время), которая дополнительно позволяет оптимизировать технологические параметры (например, скорость склеивания) и осуществить контроль качества.

### Безопасность

Робототехническая система должна соответствовать действующему на данный момент стандарту (DIN EN ISO 10218-1). Выбор возможных для использования компонентов безопасности (безопасности установки или безопасности персонала)

Критерий	Описание
Грузоподъемность/ дополнительная нагрузка	Вес инструмента, при манипулировании это всегда вес инструмента + вес заготовки  Дополнительная нагрузка. например, клапанный остров или устройство подачи проволоки
Рабочая зона	Данные рабочей зоны приведены в спецификации.  Внимание! Рабочая зона всегда определяется с помощью точки запястья
Область применения	Проверить область применения по Интернету/в каталоге продукции
Продолжительность такта	Проверять продолжительность такта рекомендуется с помощью программы KUKA.Sim

### 3.4 Конфигурация системы управления

Система управления KRC предлагает комплексную концепцию управления и привода для роботов KUKA всех классов грузоподъемности.



Рис. 3-4: Регулирование осей (V)KR C4

Кроме того, с помощью системы управления KRC и двигателей KUKA также можно управлять кинематическими системами других производителей. Для этого используется обозначение «KMC» – KUKA Motion Control (система контроля движения KUKA).

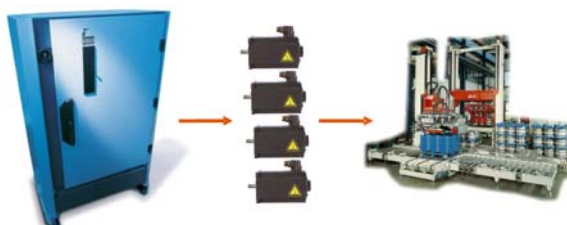


Рис. 3-5: KMC с внешней системой кинематики

### 3.5 Выбор исполнительного органа/инструмента

В современном мире термин «исполнительный орган» нашел свое применение и в технике. Здесь он используется как синоним понятия «исполнительный элемент». Так в робототехнике захваты, измерительные

инструменты, обрабатывающие инструменты и другие рабочие элементы, перемещающиеся в рабочей зоне в соответствии с заданной программой и служащие для манипулирования роботом окружающих предметов, называются концевыми исполнительными органами.

Для каждого инструмента определяется начало системы координат, Tool Center Point (TCP). Систему управления роботом можно сконфигурировать таким образом, чтобы все технические характеристики перемещения за основу брали точку TCP.

#### Возможные инструменты робота:

- Захват (кулачковый захват, вакуумный захват)
- Сварочный захват (клещи)
- Сварочная горелка
- Сопло для покрытия лаком
- Сопло для нанесения клея
- Водоструйная головка
- Оптические устройства для лазерной сварки/резки
- Сверлильная/фрезерная головка
- Винтоверт
- Инструмент для резки (пила, лезвие)
- Датчики замера

### 3.6 Выбор системы энергоснабжения

Для исполнительного органа (инструмента) на руке робота требуются энергия и сигналы управления (например, сжатый воздух, сигналы для захвата и пр.). Вид энергии зависит от типа используемого инструмента. Для подачи энергии в основном имеется две возможности:

- внешняя система подачи энергии, не зависящая от робота
- встроенная в робот система подачи энергии

#### Внешняя система подачи энергии

При использовании внешней системы подачи энергии энергия подается независимо от робота через кран-балку или похожие устройства и подвешенные к ней системы шлангов и кабелей. Однако, для данного вида подачи энергии существует риск повреждения системы шлангов и кабелей в результате перемещения робота (при их зацеплении за что-либо), особенно в фазе ввода робота в эксплуатацию. Кроме того, как правило, для этого требуются объемные системы шлангов и кабелей.



Рис. 3-6: Сварная секция для сварки в среде защитного газа с внешней системой подачи энергии

**Встроенная система подачи энергии**

При использовании встроенной системы подачи энергии необходимая энергия подается через узлы робота или вдоль них. При нормальной подаче энергии робот не может повредить системы шлангов и кабелей.



Рис. 3-7: Робот со встроенной системой энергоснабжения

**3.7 Подсоединение периферийных устройств (с помощью магистральной шины)**



Рис. 3-8: Возможности связи (V)KR C4

Для коммуникации между роботом и периферийными устройствами предлагается целый ряд возможностей:

- Встроенные аналоговые входы/выходы
- Шинные системы
  - PROFINET
  - Шина PROFIBUS
  - Внутренняя шина INTERBUS
  - ETHERNET IP
  - Can-Bus/DeviceNet
- Локальная сеть Ethernet

### 3.8 Использование датчиков

#### Датчик

Датчиком или щупом для измерений в технике называют элемент, который помимо определенных физических или химических свойств (например, теплового излучения, температуры, влажности, давления, яркости, магнитных свойств, ускорения, усилия) также может качественно или количественно (в виде значения измерения) определять структурные свойства окружающих предметов и пространства.

В качестве датчиков сигналов датчики играют значительную роль в технике во всех автоматизированных процессах. Определяемые ими значения или состояния обычно усиливаются электрически и анализируются соответствующей системой управления, которая затем определяет дальнейшие действия.

Система датчиков в робототехнической ячейке или на роботе фиксирует характеристики окружающего пространства робота:

- Состояния объектов (например, открытое или закрытое состояние)
- Возможность столкновения с препятствиями
- Физические величины в техническом процессе (например, значение усилия)
- Положение позиционных знаков и объектов
- Контур объектов
- Изображения окружающего пространства/предметов (пиксельные изображения)

### 3.9 Предохранительные устройства

При эксплуатации промышленных роботов неизбежно происходит контакт между человеком и роботом. Обычно этот контакт ограничивается фазой ввода в эксплуатацию и проведением работ по настройке и техобслуживанию. В фазе производства рабочие зоны человека и робота строго разделены. Тем не менее, даже за то короткое время, когда возможен физический контакт, в результате неправильных действий человека часто происходят серьезные несчастные случаи.

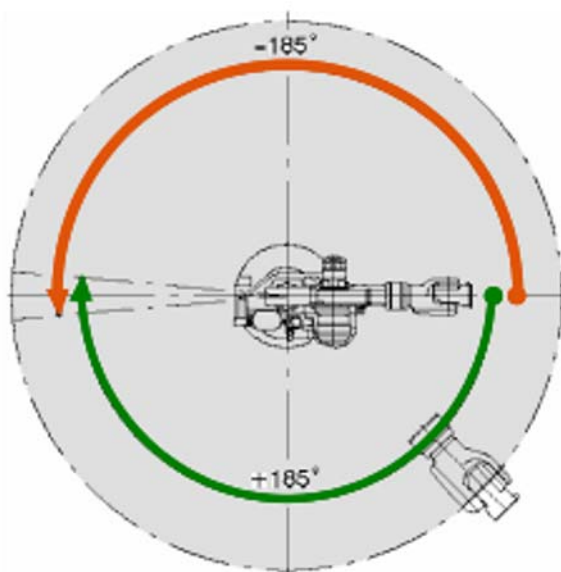
Возможные системы защиты установки и/или персонала на роботах KUKA:

- Программируемые концевые выключатели
- Механические концевые упоры
- Зонный контроль за счет разделения на рабочие зоны
- Технология Safe Robot
- Защитное ограждение
- Внешние датчики безопасности
  - Предохранительный выключатель
  - Световая завеса
  - Контактные коврики
  - Лазерные сканеры



Программируемые  
концевые  
выключатели

$\$SOFTN\_END[1] = -185^\circ$



$\$SOFTP\_END[1] = 185^\circ$

Рис. 3-9: Программные концевые выключатели положительных или отрицательных зон

Рабочая зона роботов по всем осям ограничивается программируемыми концевыми выключателями. Механически рабочие зоны осей 1, 2, 3 и 5 ограничиваются концевыми упорами с функцией амортизатора.

**i** Программируемые концевые выключатели разрешается использовать только для защиты установки.

Механические  
концевые упоры

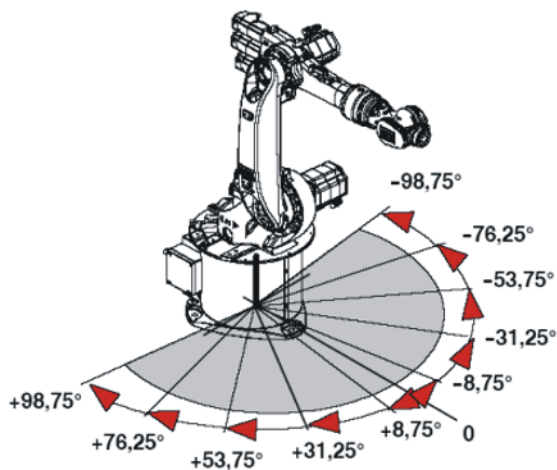
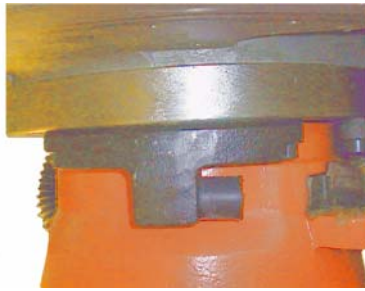


Рис. 3-10: Ограничение рабочей зоны оси 1



**Рис. 3-11: Концевой упор оси 1**

В качестве принадлежностей можно приобрести ограничители рабочей зоны. Для осей 1 - 3 они представляют собой механические упоры для независимого от конкретного производственного задания ограничения соответствующей рабочей зоны.

**i** Механические ограничители рабочей зоны можно использовать как для защиты персонала, так и для защиты установки.

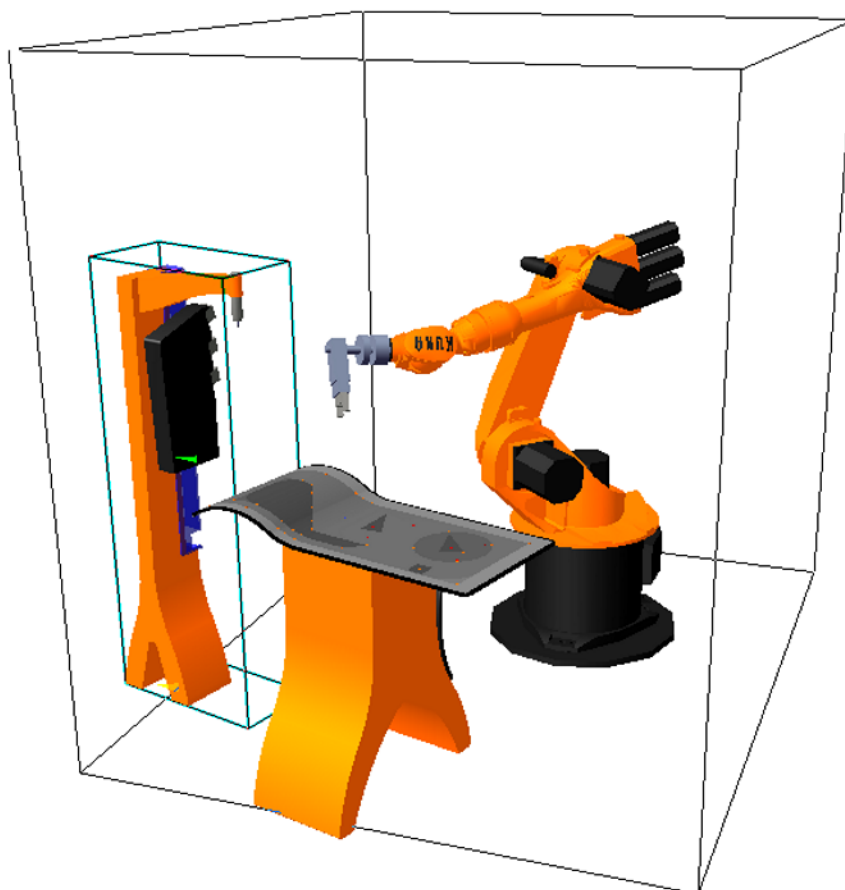
### **Зонный контроль за счет разделения на рабочие зоны**

При применении данных рабочих зон следует различать стандартные рабочие зоны в системном программном обеспечении KUKA и рабочие зоны технологии KUKA.SafeRobot:

#### **Рабочие зоны при работе с роботами KUKA**

##### **Стандартные рабочие зоны**

- Возможен автоматический контроль макс. за 8 кубическими или зависимыми от движений конкретной оси рабочими зонами. Эти рабочие зоны могут также накладываться одна на другую, образуя более сложные формы.



**Рис. 3-12: Пример декартовой рабочей зоны**

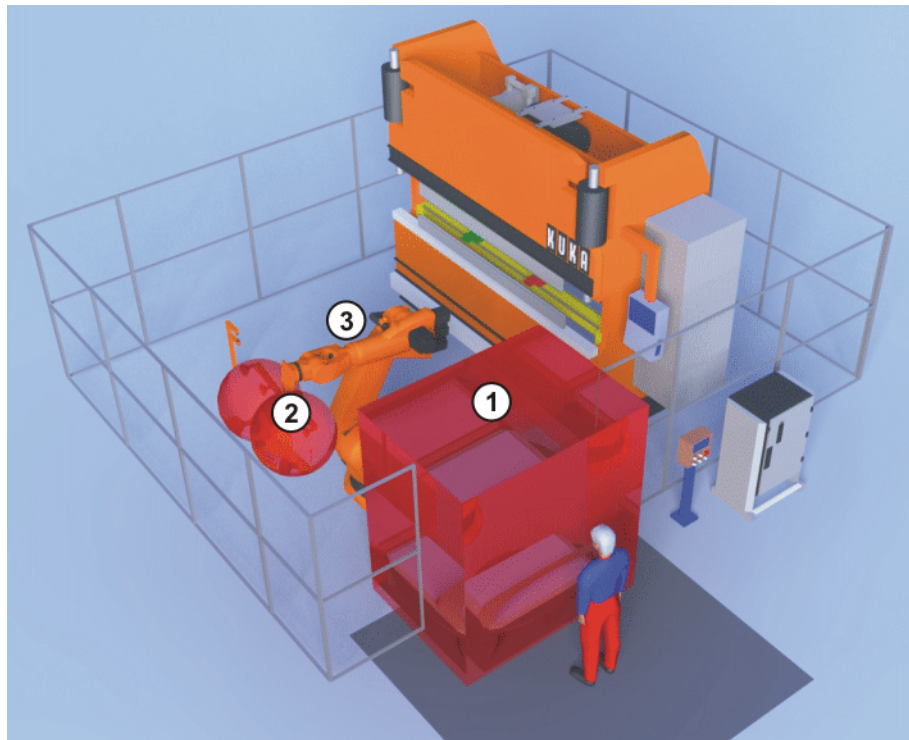
- При нарушении одной из определенных рабочих зон система управления активизирует соответствующий предварительно определенный выход. Затем подаваемый выходной сигнал может быть обработан программой KRL или внешним управляющим компьютером. При этом возможен останов робота или выдача сообщения об ошибке.



Стандартные рабочие зоны (из стандартного ПО) разрешается использовать только для защиты установки.

#### **Безопасные для людей рабочие зоны**

- 16 свободно конфигурируемых (в прямоугольной системе координат или в соответствии с движением осей) контролируемых зон (рабочих или защитных)



**Рис. 3-13: Пример защитного пространства с декартовыми координатами**

- 1 Защитная зона
- 2 Шары на инструменте
- 3 Робот

- 16 свободно конфигурируемых инструментов, моделируемых макс. с 6 шарами каждый.
- Защитные функции KUKA.SafeOperation отвечают категории 3 и уровню производительности d согласно EN ISO 13849- 1:2007. Это соответствует уровню SIL 2 в соответствии с EN 62061.



Рабочие зоны, спроектированные и проверенные с помощью SafeOperation, можно использовать как для защиты персонала, так и для защиты установки.

### Защитные ограждения



**Рис. 3-14: Повреждение защитного ограждения**

Требования к защитным устройствам более подробно описаны в **Директиве по машиностроению**, приложение I, пункт 1.4.

#### Оградительные и неоградительные защитные устройства

- Должны иметь стабильную конструкцию, например, быть ударопрочными

- Должны надежно удерживаться в своем положении
- Не должны вызывать дополнительную опасность, например, опасность заземления при закрывании, опасность удара, опасность излучения
- Не должны предоставлять возможность просто преодолеть их или сделать неэффективными
- ...

Кроме того, оградительные защитные приспособления по возможности должны защищать от выброса или падения материалов и предметов, а также от воздействия обусловленных работой машины эмиссий.



Дополнительную информацию см. в соответствующих нормативах и предписаниях.

### Внешние датчики безопасности

Если нельзя избежать ситуаций, при которых обслуживающий персонал будет находиться в зоне перемещения робота (например, при подаче деталей), необходимо защитить опасную зону контактной панелью/ковриком, световой завесой или лазерным сканером. Для защиты защитных дверей или заслонок в основном используются предохранительные выключатели.



Дополнительную информацию см. в соответствующих нормативах и предписаниях.



## 4 Промышленный робот

### 4.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Что такое робот?
- Конструкция робота
- Расположение главных осей
- Абсолютная точность и стабильность повторяемости



Рис. 4-1: Смена главы

### 4.2 Запуск робота

---

Продолжи-  
тельность: 00:10:00  
Оснащение:  
Информация:  
Сертификат:

---

**Что такое робот?** Термин *робот* образован от славянского слова *работа*, который означает *тяжелый труд*.

Официальное определение промышленного робота звучит следующим образом: «робот – свободно программируемый, управляемый программой манипулятор».

К роботу также относятся система управления и пульт управления, а также их соединительные кабели и ПО.



**Рис. 4-2: Промышленный робот**

- 1 Система управления (шкаф управления (V)KR C4)
- 2 Манипулятор (механика робота)
- 3 Пульт управления и программирования (KUKA smartPAD)

Все за пределами системы промышленного робота обозначается в качестве *периферийного оборудования*:

- инструменты (рабочие органы);
- предохранительное приспособление;
- ленты транспортеров;
- датчики;
- машины;
- и т. д.

### 4.3 Описание и конструкция

Роботы в силу своих технических характеристик относятся к классу «Модульные автоматы».

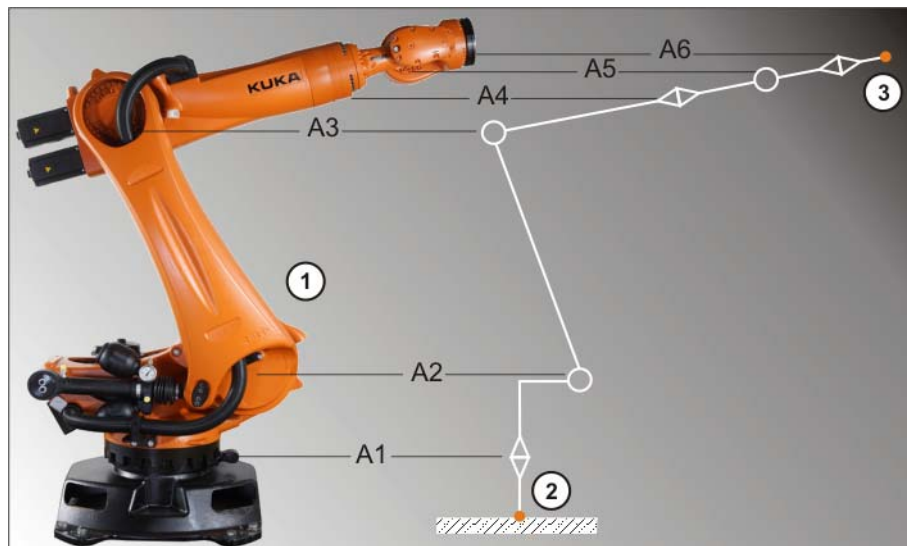


Рис. 4-3: Манипулятор

- 1 Манипулятор (механика робота)
- 2 Запуск кинематической цепочки: основание робота (ROBROOT)
- 3 Свободный конец кинематической цепочки: фланец (FLANGE)
- A1 Оси робота 1 - 6
- 
- A6

#### Директива VDI 2860

Определение согласно Директиве VDI 2860 звучит следующим образом:

Роботы – это подвижные автоматы универсального применения с несколькими осями, движения которых можно свободно программировать, задавая последовательность, маршруты или углы передвижений, и при необходимости направлять сигналами датчиков. Роботы можно оснастить захватами, инструментами или другими производственными средствами, и они могут выполнять операции по манипулированию и/или обработке.

#### EN 10218-1

Определение согласно европейскому стандарту EN 10218-1 (ранее EN 775) звучит следующим образом:

Робот/промышленный робот – это автоматически управляемый, свободно программируемый многофункциональный манипулятор, который можно запрограммировать по трем или более осям и использовать на автоматических линиях стационарно или с возможностью перемещения.

В конструкцию робота входит:

- манипулятор (включая исполнительные элементы);
- блок управления, включая переносное программирующее устройство и все коммуникационные интерфейсы (аппаратное и программное обеспечение).

Вышеперечисленное также включает в себя все дополнительные оси, которыми управляет система управления роботом.

## 4.4 Механика робота KUKA

Продолжительность:	00:20:00
Оснащение:	Модель робота
Информация:	
Сертификат:	

### Что такое манипулятор?

Манипулятор – собственная механика робота. Он состоит из определенного количества подвижных соединенных друг с другом звеньев (осей). Речь идет кинематической цепочки.

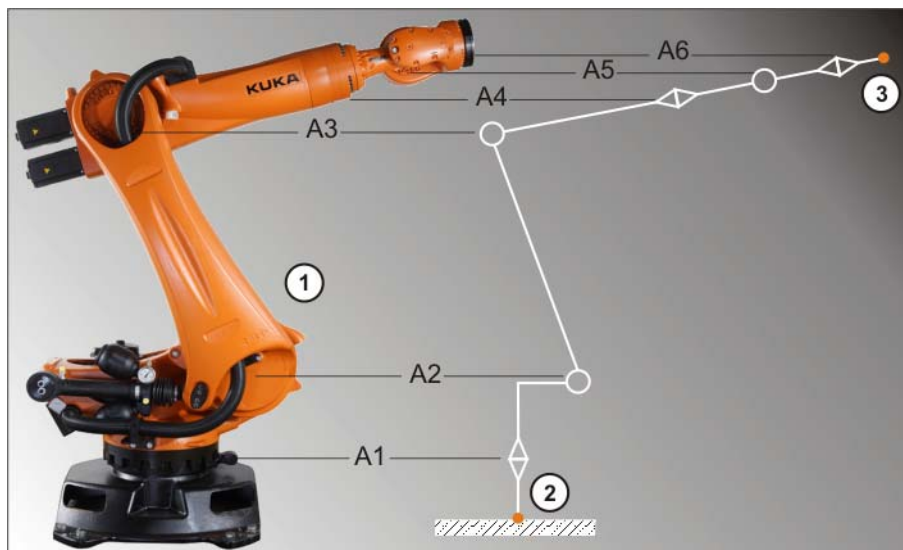


Рис. 4-4: Манипулятор

- 1 Манипулятор (механика робота)
- 2 Запуск кинематической цепочки: основание робота (ROBROOT)
- 3 Свободный конец кинематической цепочки: фланец (FLANGE)
- A1 Оси робота 1–6
- 
- A6

Перемещение отдельных осей выполняется посредством направленного регулирования серводвигателей. Они соединены через редукторы с отдельными компонентами манипулятора.

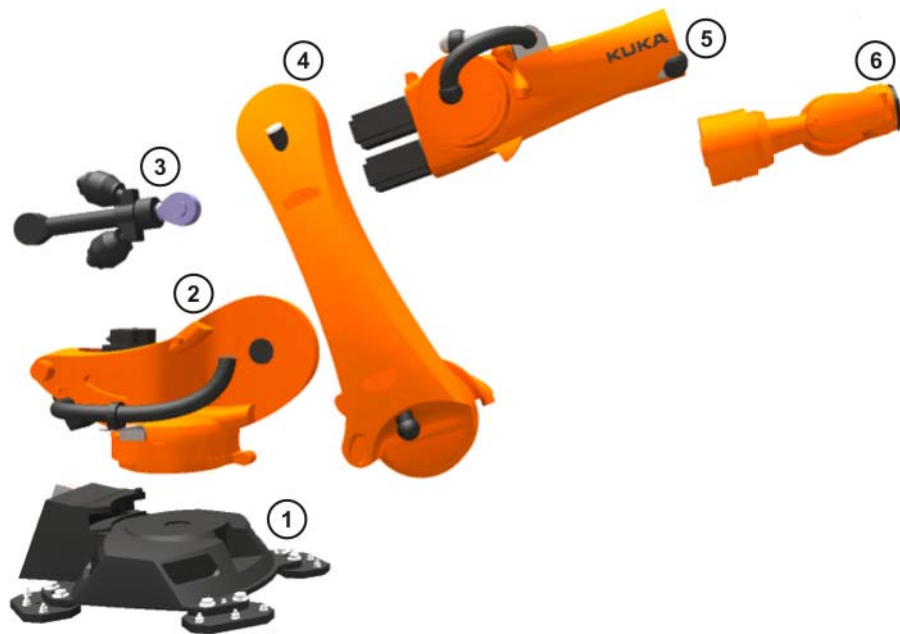


Рис. 4-5: Обзор компонентов механики робота

- |                    |               |
|--------------------|---------------|
| 1 Станина          | 4 Балансир    |
| 2 Карусель         | 5 Манипулятор |
| 3 Компенсатор веса | 6 Рука        |

Компоненты механики робота преимущественно изготовлены из алюминиевого и стального литья. В отдельных случаях также применяются компоненты из углеродного волокна.

Отдельные оси пронумерованы снизу (опора робота) вверх (фланец робота):

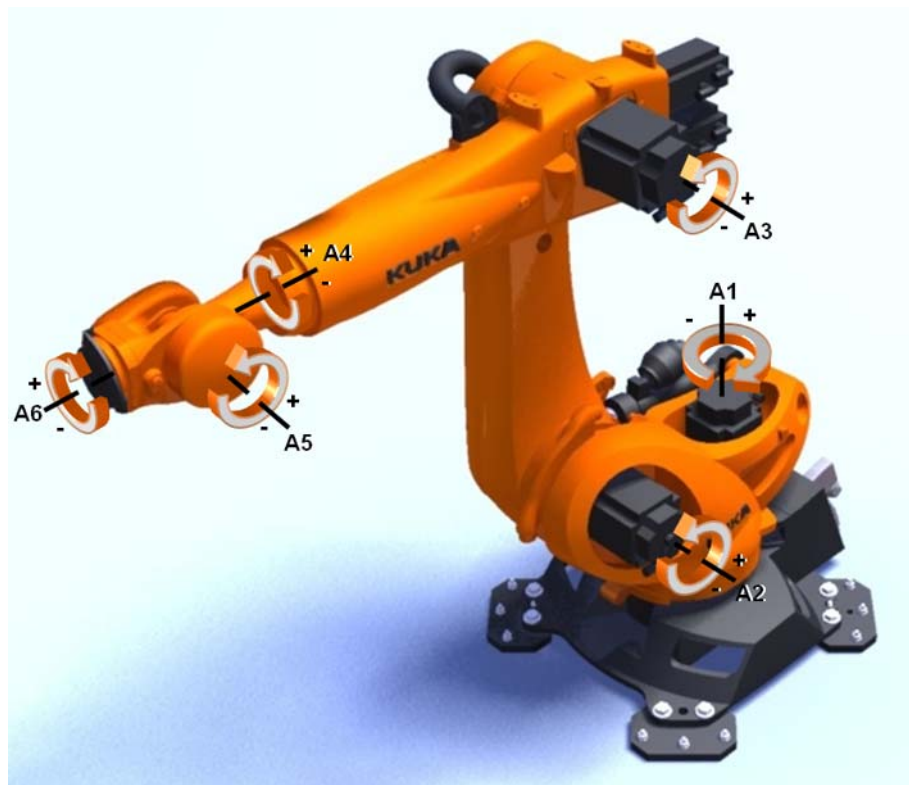


Рис. 4-6: Степени свободы робота KUKA



Выдержка из технических данных манипуляторов из ассортимента продукции KUKA:

- **количество осей:** от 4 (SCARA и роботы с параллелограммным механизмом) до 6 (стандартные вертикальные роботы с шарнирной рукой);
- **радиус действия:** от 0,35 м (KR 5 scara) до 3,9 м (KR 120 R3900 ultra K);
- **собственный вес:** от 20 кг до 4700 кг;

Стандартный робот: KR 16: 235 кг, KR 180-2 (серия 2000): 1277 кг.

- **точность:** точность повторяемости 0,015–0,2 мм.

Зоны основных осей от A1 до A3 и ручной оси робота A5 ограничены механическими концевыми упорами с амортизаторами.



На дополнительных осях можно установить отдельные концевые упоры.

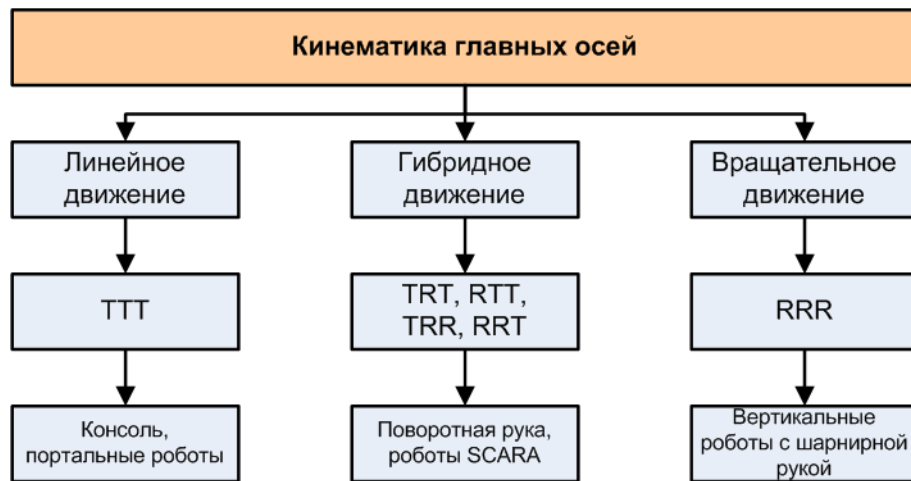
**УВЕДОМЛЕНИЕ**

При столкновении робота или одной из дополнительных осей с каким-либо препятствием или их столкновении с амортизатором робота на механическом концевом упоре устройства ограничения зоны оси существует риск повреждения робототехнической системы. Перед повторном пуске робототехнической системы в эксплуатацию следует проконсультироваться со специалистами компании KUKA Roboter GmbH. Поврежденный амортизатор необходимо немедленно заменить новым, прежде чем возобновлять эксплуатацию робота. При столкновении робота (дополнительной оси) с амортизатором со скоростью, превышающей 250 мм/с, робот (дополнительную ось) следует заменить, или же должен быть выполнен их повторный пуск в эксплуатацию специалистами компании KUKA Roboter GmbH.

#### 4.5 Расположение главных осей

Роботы различаются по типу кинематики главных осей:

- выполнение линейных перемещений (T);
- выполнение вращательных перемещений (R).



**Рис. 4-7: Кинематика главных осей**

На иллюстрации (>>> Рис. 4-8 ) объясняются различные возможности расположения главных осей и возникающие в зависимости от этого координаты:

1. Координаты формы цилиндра
2. Координаты формы шара
3. Координаты вертикальных шарниров
4. Координаты горизонтальных шарниров (роботы Scara)
5. Декартовы координаты (портальные роботы)

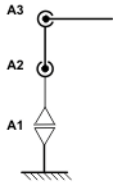
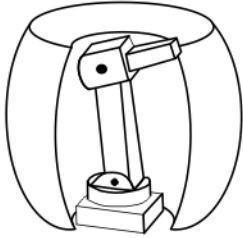
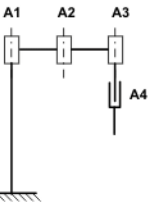
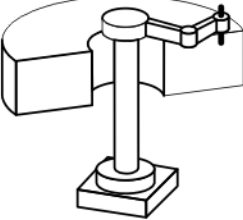
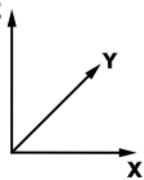
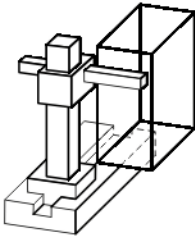
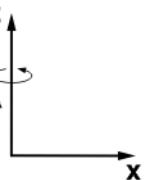
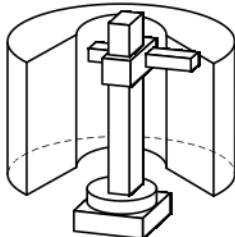
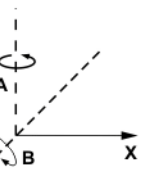

Комбинации осей	Обозначение координат	Рабочее пространство	
3 оси вращения <b>RRR</b>	Координаты шарниров (вертикальных) 	тороидальная форма	Роботы с вертикальной шарнирной рукой 
1 линейная ось 3 оси вращения <b>RRRT</b>	Координаты шарниров (горизонтальных) 	форма цилиндра	Роботы с поворотной рукой ( <b>SCARA</b> ) 
3 линейные оси <b>TTT</b>	Декартовы координаты 	прямоугольная форма	Линейные роботы Портальные роботы 
2 линейные оси 1 ось вращения <b>RTT</b>	Координаты формы цилиндра 	форма цилиндра	
1 линейная ось 2 оси вращения <b>RRT</b>	Координаты формы шара 	сферическая форма	

Рис. 4-8: Типы роботов с различным расположением осей движения (главных осей 1 - 3)

На манипуляторе робота установлены три дополнительные оси движения для позиционирования и ориентации инструмента/захвата.

Сравнение конструкций шарниров для главных осей:

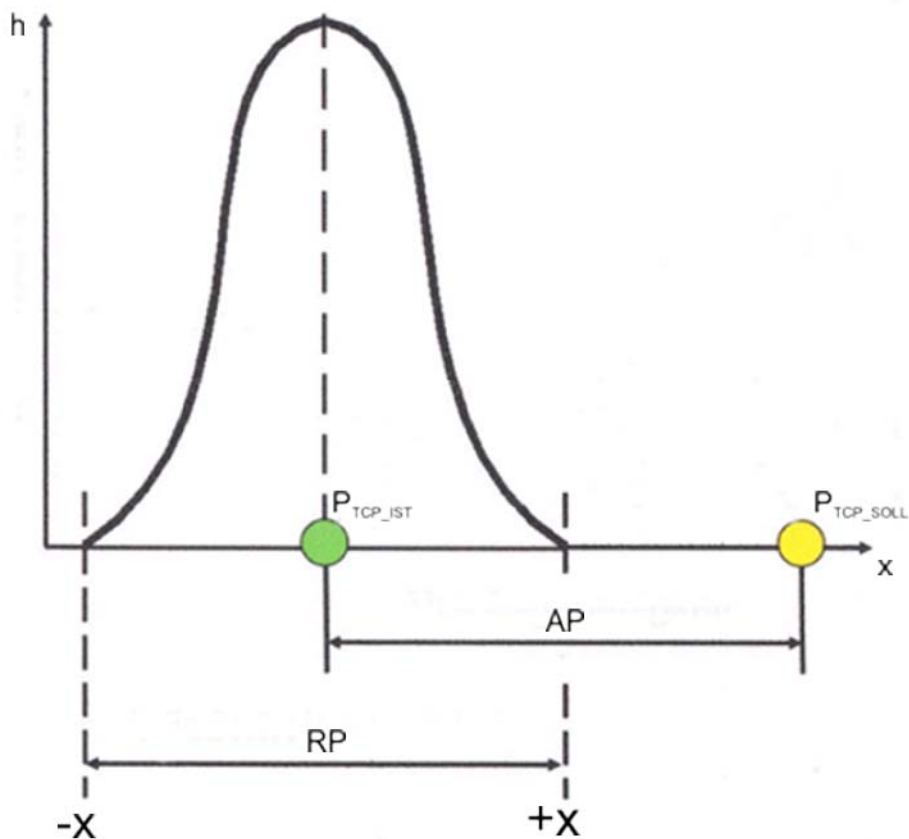
Линейные оси:	<ul style="list-style-type: none"> <li>■ свободно конфигурируемая рабочая зона</li> <li>■ удобная кинематика для операций по манипулированию и палетированию</li> <li>■ свободное увеличение рабочей зоны</li> <li>■ жесткая общая конструкция за счет механической расстыковки осей</li> </ul>
Оси вращения:	<ul style="list-style-type: none"> <li>■ быстрые движения</li> <li>■ экономично для небольших рабочих пространств</li> <li>■ выгодная кинематика для операций по обработке</li> </ul>
Принудительно состыкованные оси:	<ul style="list-style-type: none"> <li>■ экономия затрат</li> <li>■ уменьшение свободы перемещений</li> </ul>

#### 4.6 Абсолютная точность и стабильность повторяемости

Под абсолютной точностью подразумевается способность промышленного робота позиционировать свою точку Tool Center Point (TCP) из предварительно заданного направления в заданную в декартовых координатах точку пространства в пределах радиуса шара, определенного в качестве поля допуска. При этом абсолютная точность – это среднее значение отклонения точки Tool Center Point и ее ориентации во всех степенях свободы от предварительно заданной в пространстве рабочей точки и заданной ориентации (DIN EN ISO 9283).

Стабильность повторяемости – это способность промышленного робота бесконечное количество раз повторно позиционировать свою точку Tool Center Point в запрограммированную точку в рабочей зоне в пределах определенного радиуса шара с определенной ориентацией и при заданных условиях (DIN EN ISO 9283).

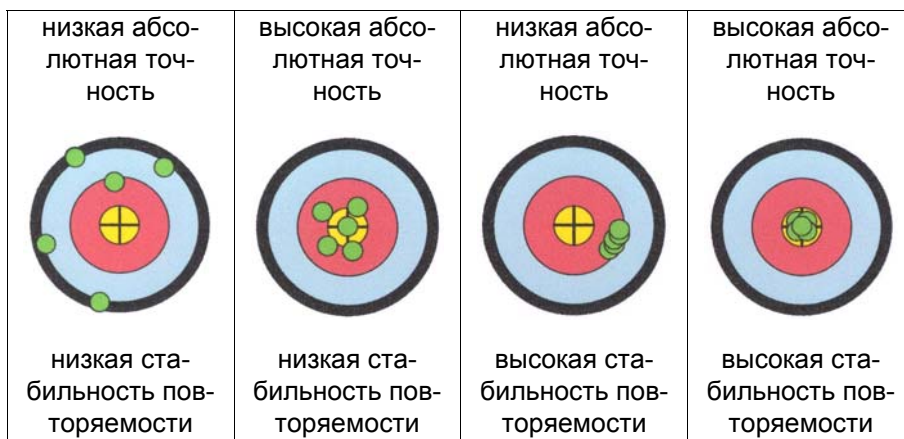
Абсолютная точность, как правило, выдерживается намного хуже, чем стабильность повторяемости.



**Рис. 4-9: Абсолютная точность и стабильность повторяемости промышленных роботов**

h	Распределение частоты
AP	Абсолютная точность
RP	Стабильность повторяемости
$P_{TCP\_IST}$	Фактическое значение положения точки TCP
$P_{TCP\_SOLL}$	Заданное значение положения точки TCP

Абсолютная точность выражает положение среднего значения цифровых, например, запрограммированных в виде текста или автономно запрограммированных позиций по отношению к заданной позиции.

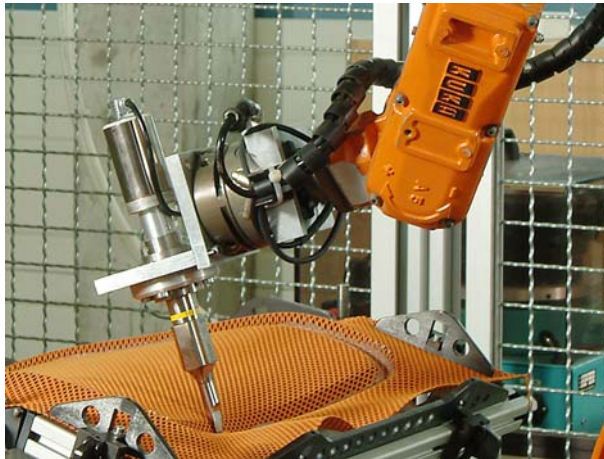


⊕ Заданная позиция

● Фактическое положение(-ия)



Стабильность повторяемости выражает отклонение отдельных фактических позиций от среднего значения фактических позиций (для сравнения: диаметр шара и центр шара).



**Рис. 4-10: Применение при нанесении клея**

Под точностью траектории подразумевается способность робота перемещать свою точку TCP по заданной в декартовой рабочей зоне траектории с определенной скоростью. Если TCP не остается в пределах определенного радиуса вокруг заданной траектории или выходит за пределы заданного диапазона скорости (заданной скорости), такая ошибка называется ошибкой отклонения.



## 5 Система управления роботом

### 5.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Обзор технических характеристик
- Компоненты системы управления роботом
- Обзор шинных систем
- Энергоэффективность



Рис. 5-1: Смена главы

#### Основные характеристики

Тип шкафа	KR C4
Количество осей	макс. 8
Вес (без трансформатора)	150 кг
Класс защиты	IP 54
Уровень шума по DIN 45635-1	в среднем 67 дБ (A)
Последовательное соединение с охладителем или без него	сбоку, расстояние 50 мм
Нагрузка на верхнюю часть при равномерном распределении	1 500 Н

#### Подключение к сети

Систему управления роботом допускается подключать только к сети с заземленной нулевой точкой.

Если заземленная нулевая точка отсутствует, или напряжение сети не соответствует указанным здесь параметрам, следует использовать трансформатор.

Номинальное напряжение питающей сети, по выбору	3x380 В перем. тока, 3x400 В перем. тока, 3x440 В перем. тока или 3x480 В перем. тока
Допуск для номинального напряжения питающей сети	Номинальное напряжение питающей сети $\pm 10\%$
Сетевая частота	49 ... 61 Гц
Полное сопротивление сети до точки подключения системы управления роботом	$\leq 300$ мОм
Ток полной нагрузки	см. заводскую табличку
Предохранитель со стороны питающей сети без разделительного трансформатора	не менее 3x25 А, инерционные

Предохранитель со стороны питающей сети с разделительным трансформатором	не менее 3x32 А, инерционные при 13 кВА
Выравнивание потенциалов	Общей нулевой точкой для кабелей выравнивания потенциалов и всех защитных кабелей является стандартная шина энергоблока.

**Условия окружающей среды**

Температура окружающей среды при работе без охладителя	+5 ... 45 °C (278– 318 К)
Температура окружающей среды при работе с охладителем	+20 ... 50 °C (293– 323 К)
Температура окружающей среды при хранении и транспортировке с аккумуляторами	-25 ... +40 °C (248– 313 К)
Температура окружающей среды при хранении и транспортировке без аккумуляторов	-25 ... +70 °C (248– 343 К)
Изменение температуры	макс. 1,1 К/мин
Класс влажности	3к3 согласно DIN EN 60721-3-3; 1995
Высота установки	<ul style="list-style-type: none"> <li>■ до 1000 м в.у.м. без снижения производительности;</li> <li>■ от 1000 м до 4000 м в.у.м. со снижением производительности на 5 %/1000 м</li> </ul>

**УВЕДОМЛЕНИЕ**

Во избежание глубокой разрядки и повреждения аккумуляторов их необходимо регулярно заряжать в зависимости от температуры их хранения.  
 При температуре хранения +20 °C или меньше аккумуляторы следует заряжать через каждые 9 месяцев.  
 При температуре хранения от +20 °C до +30 °C аккумуляторы следует заряжать через каждые 6 месяцев.  
 При температуре хранения от +30 °C до +40 °C аккумуляторы следует заряжать через каждые 3 месяца.

**Виброустойчивость**

Вид нагрузки	При транспортировке	При продолжительной эксплуатации
Эффективное значение ускорения (непрерывные колебания)	0,37 г	0,1 г
Диапазон частоты (непрерывные колебания)	4..120 Гц	
Ускорение (удар в направлении X/Y/Z)	10 г	2,5 г
Форма кривой продолжительности (удар в направлении X/Y/Z)	Полу-синус/11 мс	

Если предполагается более интенсивная механическая нагрузка, систему управления следует установить на амортизаторы.

**Блок управления**

Напряжение питания	27,1 В ± 0,1 В пост. тока
--------------------	---------------------------

**Управляющий компьютер**

Главный процессор	см. в актуальном комплекте поставки
Модуль памяти DIMM	см. в актуальном комплекте поставки (мин. 2 ГБ)
Жесткий диск	см. в актуальном комплекте поставки

**Пульт управления KUKA smartPAD**

Напряжение питания	20–27,1 В пост. тока
Габариты (ШхВхГ)	ок. 33x26x8 см <sup>3</sup>
Экран	Сенсорный цветной экран 600 x 800 точек
Диагональ экрана	8,4 "
Интерфейсы	USB
Вес	1,1 кг

**Длина кабелей**

В инструкции по эксплуатации, инструкции по монтажу манипулятора и/или инструкции по монтажу и эксплуатации KR C4, в разделе о внешней кабельной проводке для систем управления роботами, указаны обозначения кабелей, длина кабелей (стандартная), а также специальная длина.



При применении кабельных удлинителей пульта управления smartPAD разрешается использовать только два удлинителя. Общая длина кабеля не должна превышать 50 м.



Длина кабелей в отдельных каналах блока RDC не должна отличаться более чем на 10 м.

**5.2 Габаритные размеры системы управления роботом**

На изображении (>>> Рис. 5-2 ) показаны габаритные размеры системы управления роботом.

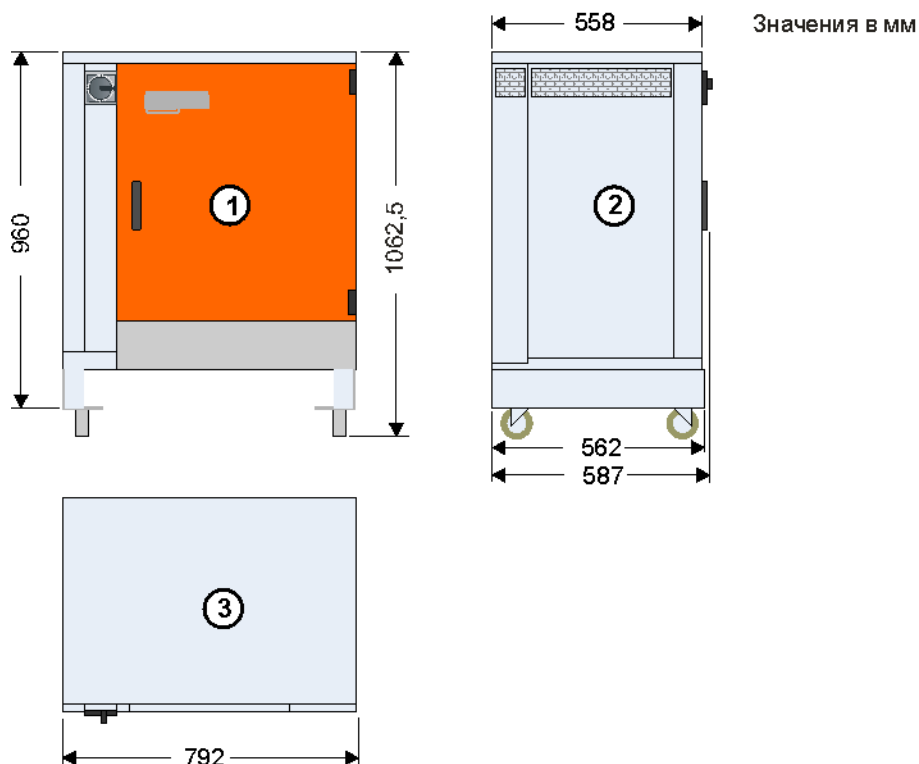


Рис. 5-2: Габариты

- 1 Вид спереди
- 2 Вид сбоку
- 3 Вид сверху

### 5.3 Минимальные расстояния для системы управления роботом

На изображении (>>> Рис. 5-3 ) показаны выдерживаемые минимальные размеры системы управления роботом.

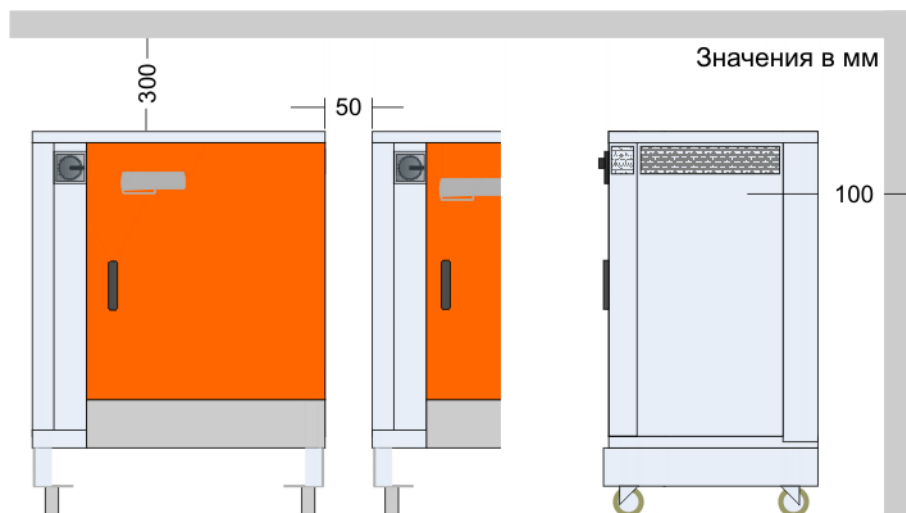


Рис. 5-3: Минимальные расстояния

**УВЕДОМЛЕНИЕ** Несоблюдение минимальных расстояний может привести к повреждению системы управления роботом. Указанные минимальные расстояния должны неукоснительно соблюдаться.



**i** Работы по техническому обслуживанию и ремонту системы управления роботом проводятся сбоку или сзади. Для этого необходим доступ к системе управления роботом. Если боковая и задняя стенки недоступны, следует по возможности установить систему управления роботом в положение, в котором работы могут быть выполнены.

#### 5.4 Обзор системы управления роботом

KR C4 вид  
спереди

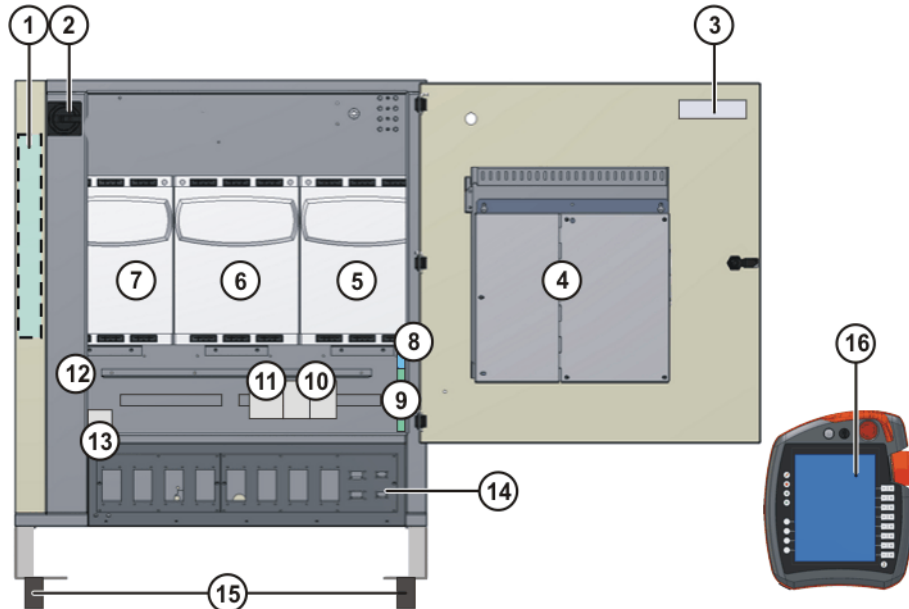
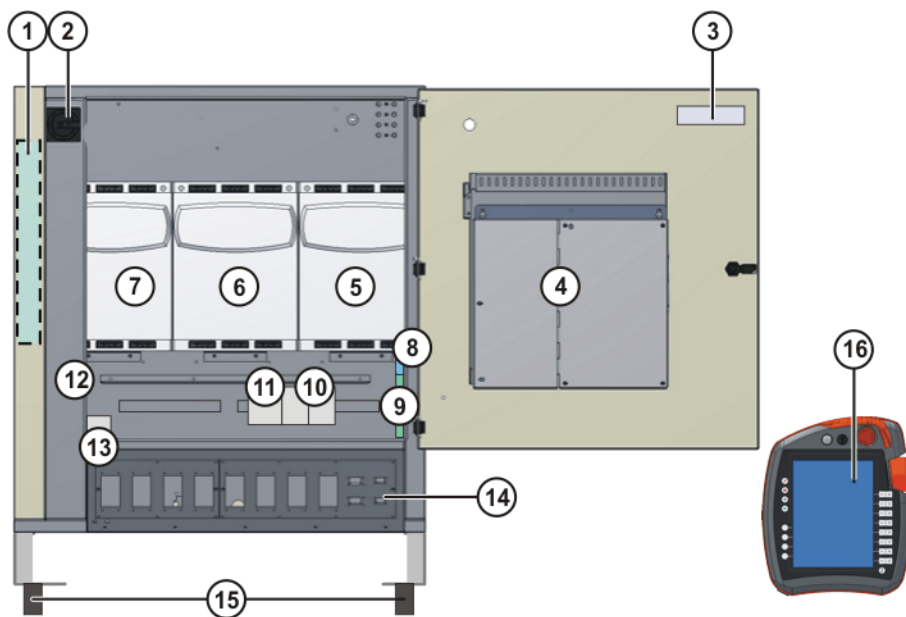


Рис. 5-4: Обзор системы управления роботом (вид спереди)

- |   |  |    |                           |
|---|--|----|---------------------------|
| 1 | Сетевой фильтр                                     | 9  | CCU                       |
| 2 | Главный выключатель                                | 10 | Контактор                 |
| 3 | CSP  | 11 | Сетевой коммутатор        |
| 4 | Управляющий ПК                                     | 12 | Предохранительный элемент |
| 5 | Сетевой блок питания привода с регулятором привода | 13 | Аккумуляторы              |
| 6 | Регулятор привода осей 4–6                         | 14 | Панель подсоединения      |
| 7 | Регулятор привода осей 1–3                         | 15 | Корпус                    |
| 8 | Тормозной фильтр                                   | 16 | smartPAD                  |

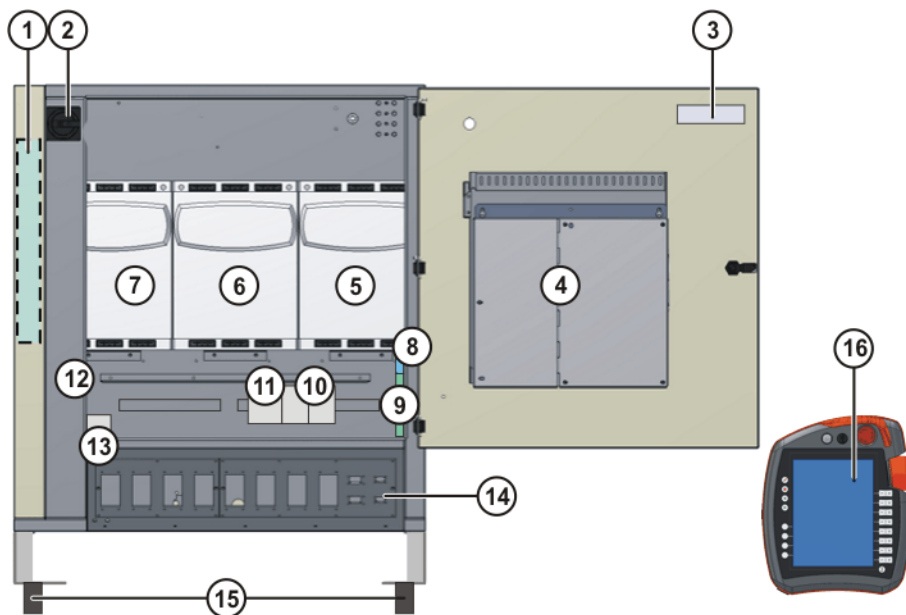
**KR C4 вид  
спереди**



**Рис. 5-5: Обзор системы управления роботом (вид спереди)**

- 1 Сетевой фильтр
- 2 Главный выключатель
- 3 Пульт системы контроллеров (CSP)

**KR C4 вид  
спереди**



**Рис. 5-6: Обзор системы управления роботом (вид спереди)**

- 4 Управляющий ПК
- 5 Сетевой блок питания привода с регулятором привода
- 7 Регулятор привода осей 1–3

KR C4 вид  
спереди

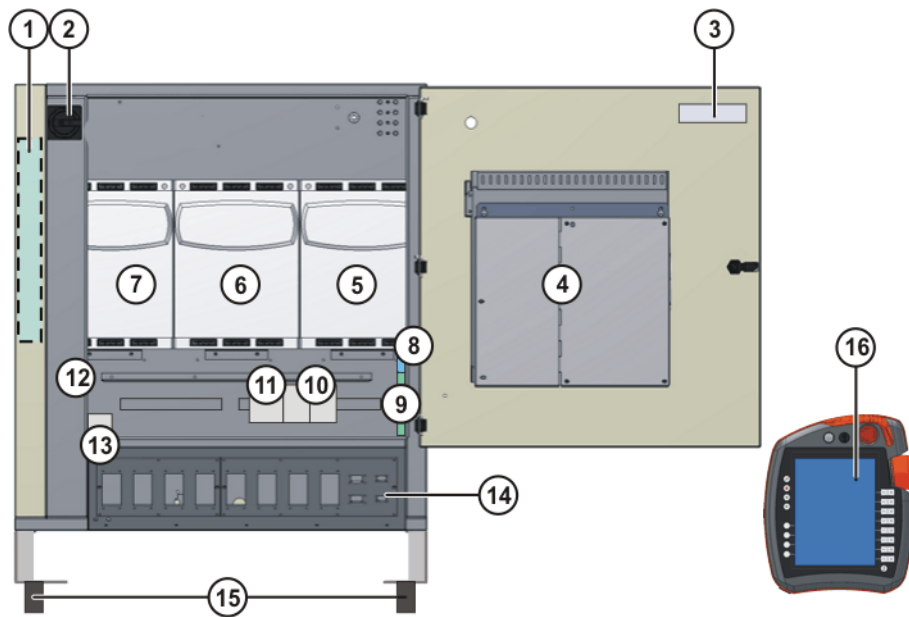


Рис. 5-7: Обзор системы управления роботом (вид спереди)

- 6 Регулятор привода осей 4–6
- 8 Тормозной фильтр
- 9 Шкаф управления (CCU)

KR C4 вид  
спереди

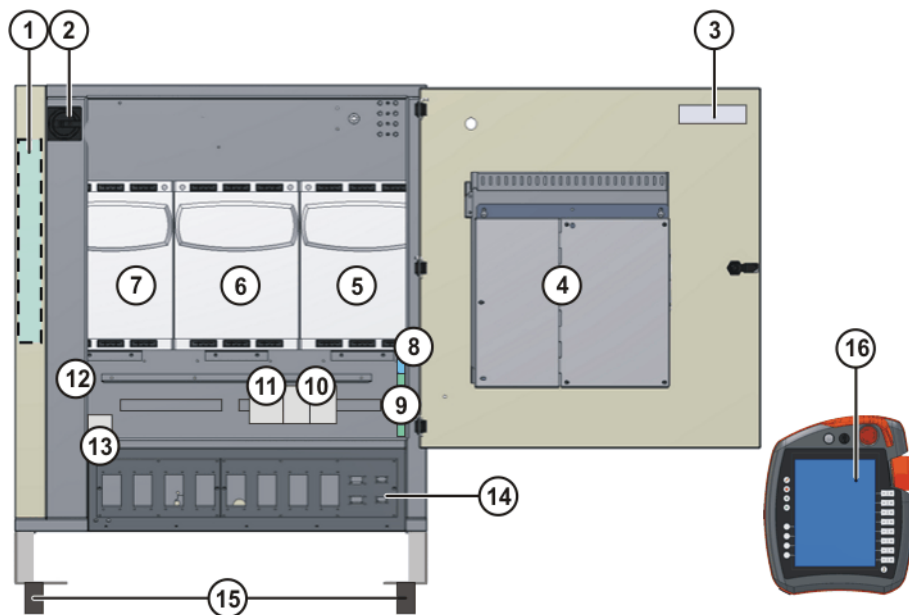
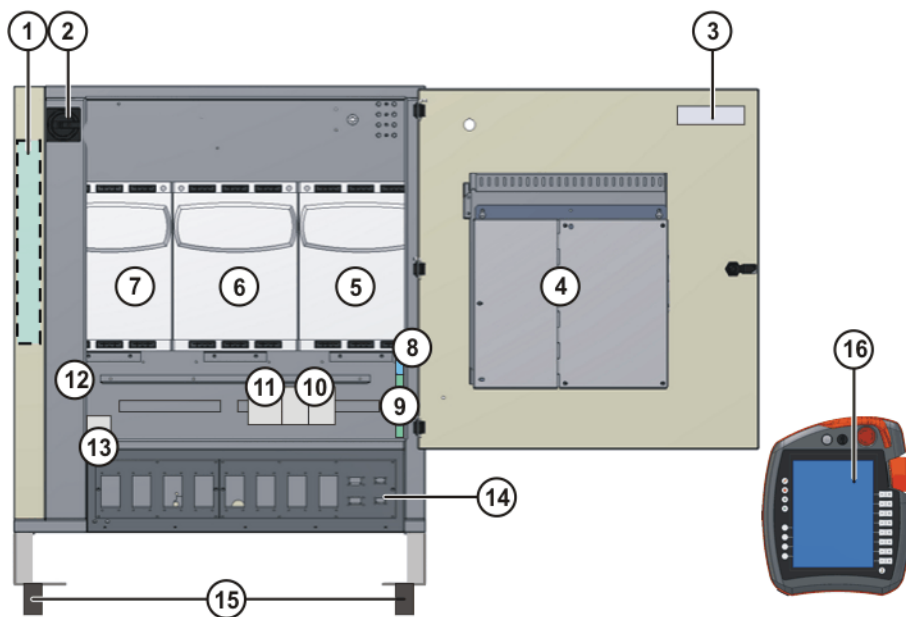


Рис. 5-8: Обзор системы управления роботом (вид спереди)

- 10 Контакттор
- 11 Сетевой коммутатор
- 12 Предохранительный элемент

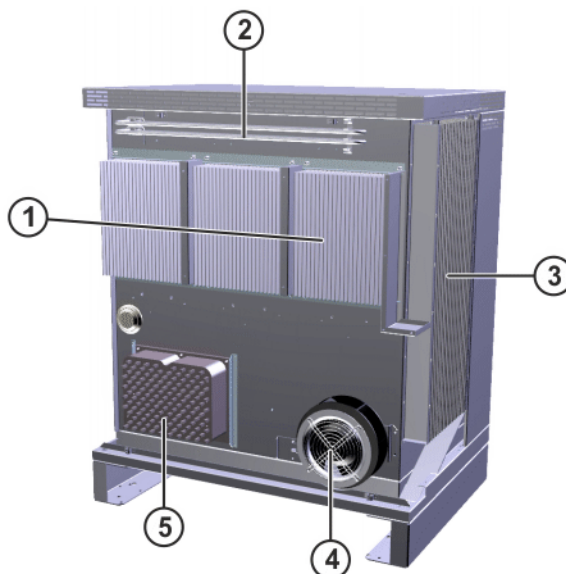
**KR C4 вид  
спереди**



**Рис. 5-9: Обзор системы управления роботом (вид спереди)**

- 13 Аккумуляторы
- 14 Панель подсоединения
- 15 Корпус
- 16 smartPAD

**Вид сзади KR C4**



**Рис. 5-10: Обзор системы управления роботом (вид сзади)**

- 1 Балластные резисторы
- 2 Теплообменник
- 3 Внешний вентилятор
- 4 Низковольтный сетевой блок питания

## 5.5 Обзор приложений и шинных систем

### Обзор приложений

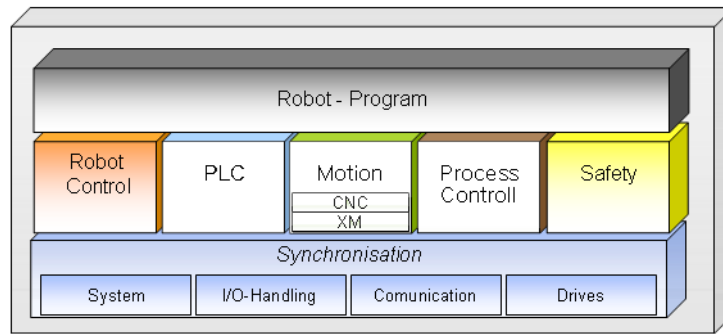


Рис. 5-11: Обзор стоечной модели

- Чтобы из программ робота получить доступ к интерфейсам аппаратного обеспечения конкретного шкафа управления, на системе управления должны быть установлены разные приложения и опциональные функции.
- К **системообразующим приложениям** относятся:
  - **RC (Robot Control)**  
Базовая система KUKA для управления роботом
  - **Safety**  
Собственная интегрируемая система управления KUKA, обеспечивающая безопасность

К **выбираемым заказчиком опциям** относятся:

- **ПЛК**  
Интегрируемый программный ПЛК для общей системы управления процессом
- **XM (eXtended Motion)**  
Интегрируемая система поддержки выполнения для библиотеки KUKA MotionControl
- **Process Control**  
Общая платформа для интеграции систем управления процессом, например, интегрируемой системы Vision

### Обзор шинных систем

- В системе управления KR C4 имеются четыре различные шинные системы на базе EtherNet.
- Каждая из этих шинных систем соединяет друг с другом различные компоненты систем управления.

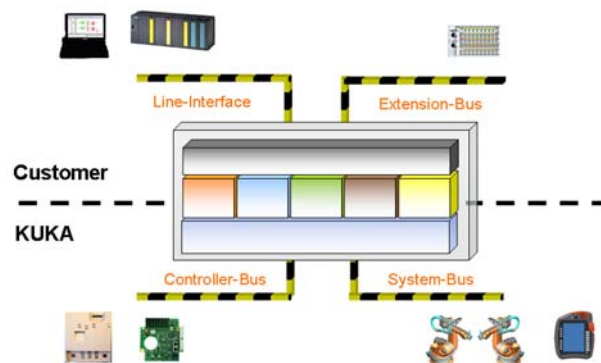


Рис. 5-12: Обзор шинных систем KR C4

Шина	Описание
<b>KCB</b> (шина контроллера KUKA)	Присоединение участника контура привода: <ul style="list-style-type: none"> <li>■ преобразователя сигнала синус.-косинус. преобразователя в цифровую форму (Resolver Digital Converter);</li> <li>■ блока питания KPP (KUKA Power Pack);</li> <li>■ регулятора KSP (KUKA System Pack);</li> <li>■ электронного наладочного устройства (Electronic Mastering Device).</li> </ul>
<b>KSB</b> (системная шина KUKA)	Для присоединения следующих систем: <ul style="list-style-type: none"> <li>■ smartPAD</li> <li>■ интерфейсной платы безопасности SIB</li> <li>■ расширенной платы SIB</li> <li>■ комплекса роботов RoboTeam</li> <li>■ других опций KUKA</li> </ul>
<b>KEB</b> (шина расширений KUKA)	Для присоединения следующих систем: <ul style="list-style-type: none"> <li>■ вводов/выводов EtherCat</li> <li>■ межсетевого шлюза для PROFIBUS</li> <li>■ межсетевого шлюза для DeviceNet</li> </ul>
<b>KLI</b> (линейный интерфейс KUKA)	Для присоединения следующих систем: <ul style="list-style-type: none"> <li>■ ПЛК</li> <li>■ подсоединения магистральной шины                             <ul style="list-style-type: none"> <li>■ PROFINET &amp; PROFI-safe</li> <li>■ EtherNet/IP &amp; CIP safety</li> </ul> </li> <li>■ подсоединение сети через TCP/IP:                             <ul style="list-style-type: none"> <li>■ архивирование данных</li> <li>■ диагностика</li> <li>■ VRP (Virtual Remote Pendant)</li> <li>■ и т. д.</li> </ul> </li> </ul>

**Подробный вид шины:**

**Главная плата D3075-K**



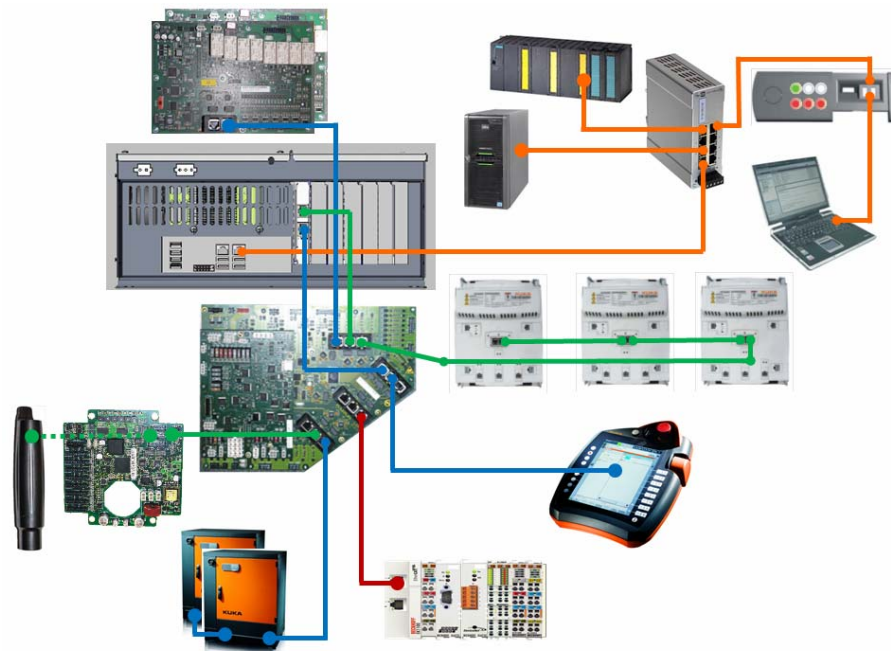


Рис. 5-13: Шинная структура

**i** В зависимости от версии главной платы подсоединения KLI и KSB могут быть поменяны местами.

**Главная плата D2608-K**

- Подсоединение KLI на сетевой карте DualNIC
- Подсоединение KSB на главной плате

**Главная плата D3076-K**

- Подсоединение KLI на главной плате
- Подсоединение KSB на сетевой карте DualNIC

### 5.5.1 Шина контроллера KUKA, КСВ

**Описание** Обзор шины контроллера KUKA:

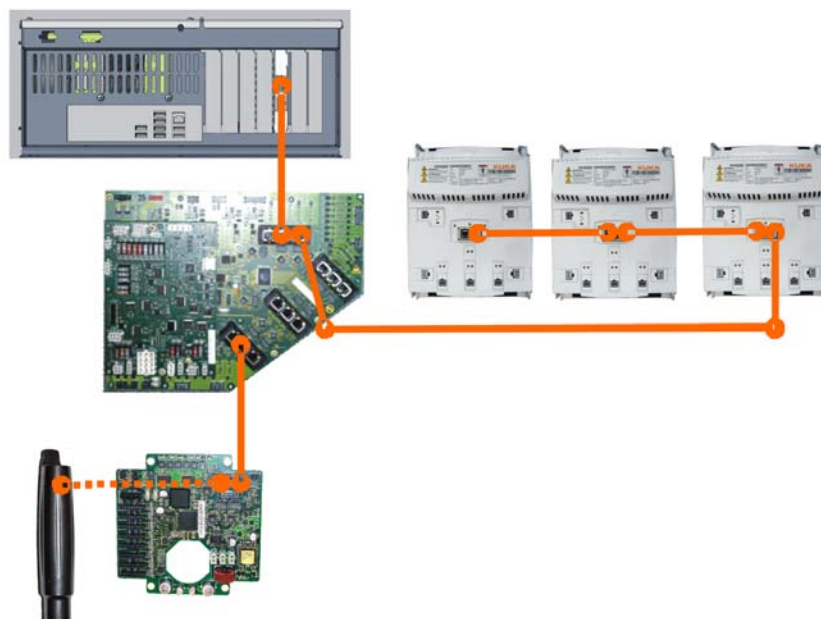


Рис. 5-14: Структура шины KCB

**Основные данные KCB**

- Шина привода на базе EtherCAT
- Время цикла 125 микросекунд
- FSOE (**F**ail **S**afe **O**ver **E**ther**C**at)

**К шине KCB относятся следующие устройства:**

- блок питания KUKA Power Pack (KPP);
- регулятор KSP A1 - 3 -KUKA Servo Pack;
- регулятор KSP A4 - 6 -KUKA;
- преобразователь сигнала синус.-косинус. преобразователя в цифровую форму (Resolver Digital Converter – RDC);
- электронное наладочное устройство (EMD) (соединяемый участник)

**5.5.2 Системная шина KUKA, KSB**

**Описание**                      Обзор системной шины KUKA:



Рис. 5-15: Структура шины KSB

**УВЕДОМЛЕНИЕ** В зависимости от версии используемой главной платы КРС для KLI и KSB могут использоваться другие интерфейсы аппаратного обеспечения.

**Главная плата D2608-K**

- Подсоединение KLI на сетевой карте DualNIC
- Подсоединение KSB на главной плате

**КРС с главной платой D3076-K**

- Подсоединение KLI на главной плате
- Подсоединение KSB на сетевой карте DualNIC

**Основные данные KSB**

- Шина на базе EtherCat
- Время цикла 1 миллисекунда
- FSOE (Fail Safe Over EtherCat)

**К KSB подключаются следующие устройства:**

- KUKA smartPAD (HMI или RDP)
- RoboTeam (через соединительный кабель)
- Безопасное соединение; SIB (интерфейсная плата безопасности) X11, X13

### 5.5.3 Шина расширений KUKA Extension Bus, KEB

**Описание** Обзор шины расширений KUKA:



Рис. 5-16: Структура шины КЕВ

**Основные данные КЕВ**

- Ведущее устройство EtherCAT Master на ССУ
- Время цикла 1 миллисекунда
- Мощная магистральная шина типа DeviceNet для простого встраивания дискретных вводов/выводов заказчика (не требуются безопасные входы/выходы)
- Конфигурирование с помощью WorkVisual

**К КЕВ подключаются следующие устройства:**

- Модули входа/выхода EtherCAT по техническим условиям заказчика
- Решения по межсетевым шлюзам для Profibus и DeviceNet

**5.5.4 Линейный интерфейс KUKA Line Interface, KLI**

**Описание**

Обзор линейного интерфейса KUKA:

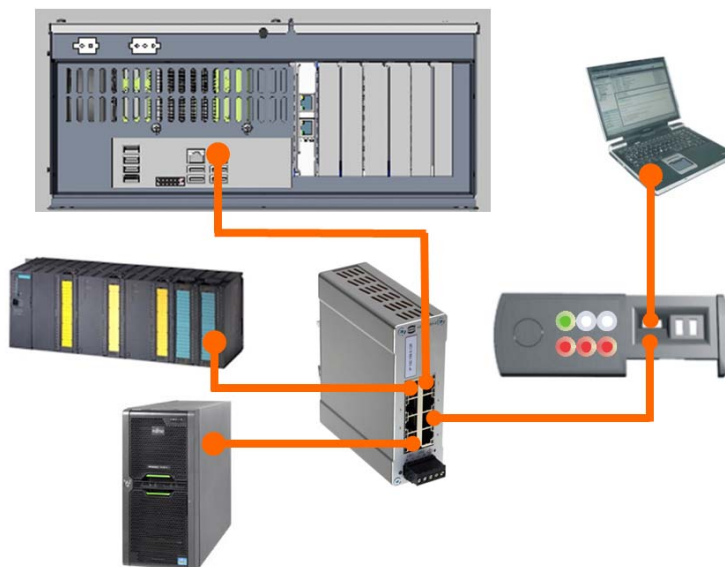


Рис. 5-17: Структура шины KLI

**УВЕДОМЛЕНИЕ**

В зависимости от версии ПО KSS подключения KLI и KSB могут быть поменяны местами.

**Главная плата D2608-K**

- Подсоединение KLI на сетевой карте DualNIC
- Подсоединение KSB на главной плате

**Главная плата D3076-K**

- Подсоединение KLI на главной плате
- Подсоединение KSB на сетевой карте DualNIC

**Основные данные KLI**

- Интерфейс заказчика X66, X67 на базе Ethernet
- Подсоединение к установке и вышестоящим уровням (сеть заказчика, сервер)
- Магистральные шины на базе Ethernet (PROFINET, PROFI-safe, EtherNet/IP, CIP Safety)
- Стандартный Ethernet (например, для архивирования и обмена данными)
- Конфигурирование с помощью WorkVisual

**К KLI подключаются следующие устройства:**

- сервисный ноутбук
- модули входа/выхода заказчика, ПЛК, ПЛК безопасности
- сервер, компьютер пульта управления

**5.6 Эффективность использования энергии**

<b>Энергосбережение с помощью KR C4</b>	<p>Особое внимание в KR C4 уделяется потреблению энергии в производстве и в режиме ожидания.</p> <p>Энергосберегающая концепция охлаждения</p> <ul style="list-style-type: none"> <li>■ Внешний контур охлаждения имеет терморегулируемый вентилятор 24 В, зависимый от нагрузки и температуры окружающей среды.</li> <li>■ Отказ внутреннего вентилятора шкафа -&gt; обеспечение циркуляции с помощью вентилятора ПК и целенаправленного подвода воздуха -&gt; усиленное использование поверхности для охлаждения -&gt; охлаждение блока питания 27 В с помощью внешнего контура охлаждения.</li> <li>■ Уменьшенная установленная мощность охлаждения и пониженная частота вращения (прибл. 220 Вт в режиме ожидания).</li> </ul>
<b>KUKA «Режимы эффективного использования энергии»</b>	<ul style="list-style-type: none"> <li>■ Режим шины привода           <ul style="list-style-type: none"> <li>■ Дезактивация шины привода (тормоза задействованы)</li> <li>■ ПК KUKA с системой управления, обеспечивающей безопасность, и соединением PROFINET еще активны</li> <li>■ Готовность к пуску через прибл. 10 сек.</li> <li>■ Потребление прибл. 158 Вт</li> </ul> </li> <li>■ Спящий режим           <ul style="list-style-type: none"> <li>■ Дезактивация всех систем в системе управления (за исключением Ethernet «Low Level»)</li> <li>■ ПК KUKA-PC с системой управления, обеспечивающей безопасность, и соединением PROFINET в «Спящем режиме»</li> <li>■ Готов к пуску через 30 - 60 сек. (в зависимости от приложения)</li> <li>■ Потребление прибл. 40 Вт</li> </ul> </li> </ul>
<b>Активация</b>	<p>Активация/дезактивация по команде на уровне установки или управляющее воздействие на систему управления</p> <ul style="list-style-type: none"> <li>■ PROFINET: подача и перемещение стандартные через AIDA и PNO (профиль ProfiEnergy на уровне SPS)</li> <li>■ Ethernet: Активация с помощью пакета Magic</li> <li>■ Вручную в управлении на месте           <ul style="list-style-type: none"> <li>■ Режим пониженного энергопотребления в пункте меню на KUKA smartPAD</li> <li>■ Таймерное включение главным выключателем</li> </ul> </li> </ul>

AIDA	Инициатива по стандартизации Немецкого союза производителей оборудования для сервисного обслуживания автомобилей
PNO	Организация пользователей Profibus



## 6 Перемещение робота

### 6.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Чтение и интерпретация сообщений
- Выбор и настройка режимов работы
- Перемещение осей робота по отдельности
- Перемещение робота в универсальной системе координат
- Перемещение робота в системе координат инструмента
- Перемещение робота в основной системе координат



Рис. 6-1: Смена главы


### 6.2 Переносное программирующее устройство KUKA smartPAD

#### 6.2.1 Передняя сторона

##### Функция

Пульт smartPAD – переносное программирующее устройство для промышленных роботов. Пульт smartPAD имеет все возможности управления и индикации, необходимые для управления промышленным роботом и его программирования.

На пульте smartPAD расположен сенсорный экран: интерфейсом smartHMI можно управлять с помощью пальца или стилуса. Внешняя мышь или клавиатура не требуется.

 В настоящей документации пульт управления smartPAD часто обозначается общим названием КСР (KUKA Control Panel, пульт управления KUKA).

Обзор

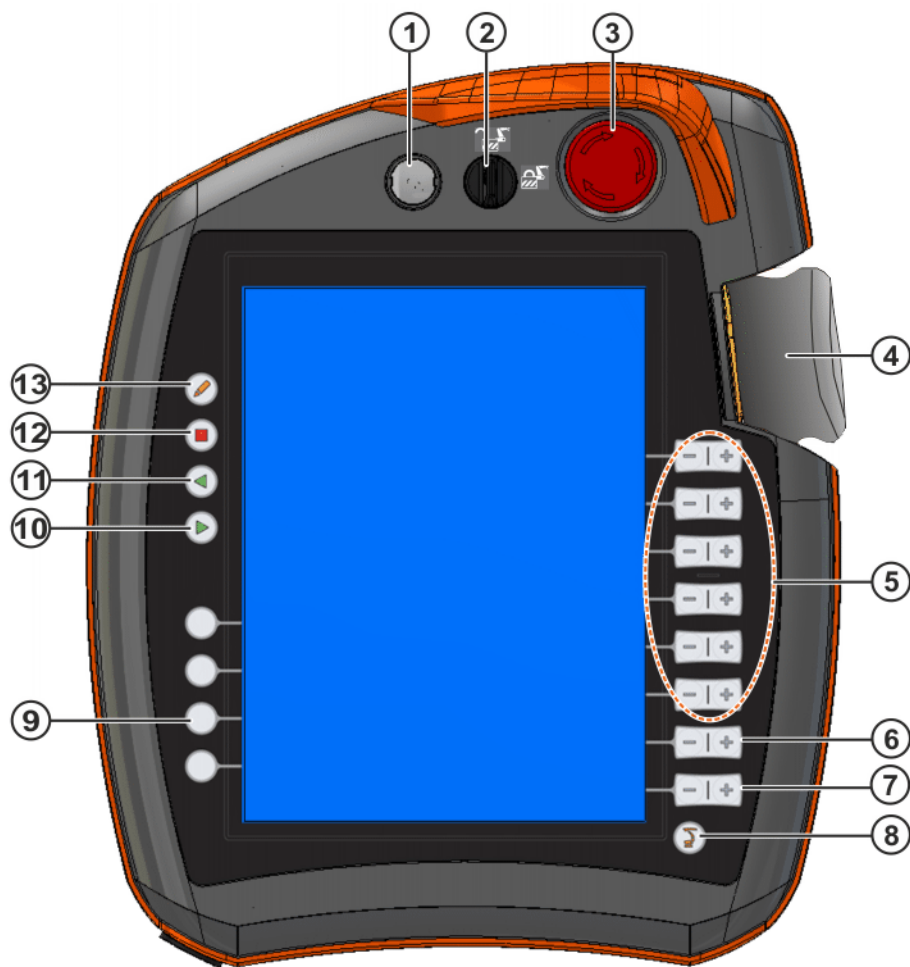


Рис. 6-2: Передняя сторона пульта управления KUKA smartPAD

Поз.	Описание
1	Кнопка для отсоединения пульта управления smartPAD.
2	Замок-выключатель для вызова менеджера соединений. Переключатель можно установить в другое положение только при вставленном ключе. С помощью менеджера соединений можно изменить режим работы.
3	Устройство АВАРИЙНОГО ОСТАНОВА. Служит для останова робота в опасных ситуациях. Устройство АВАРИЙНОГО ОСТАНОВА при нажатии блокируется.
4	Пространственная мышь: Служит для перемещения робота вручную.
5	Клавиши перемещения: служат для перемещения робота вручную.
6	Кнопка для установки программной коррекции.
7	Кнопка для установки ручной коррекции.

Поз.	Описание
8	Кнопка вызова главного меню: выводит пункты меню на экран smartHMI.
9	Клавиши состояния. Клавиши состояния в основном служат для настройки параметров из пакетов технологий. Их функция зависит от установленных пакетов технологий.
10	Клавиша пуска: клавишей пуска запускается программа.
11	Клавиша обратного пуска: Клавишей обратного пуска запускают программу в обратном направлении. Программа обрабатывается поэтапно.
12	Клавиша СТОП: клавиша СТОП останавливает работающую программу.
13	Клавиша клавиатуры: Эта клавиша выводит на экран клавиатуру. Обычно клавиатуру не нужно вызывать отдельно: интерфейс smartHMI определяет, когда требуется ввод данных с помощью клавиатуры, и она автоматически появляется на экране.

## 6.2.2 Обратная сторона

### Обзор



Рис. 6-3: Обратная сторона пульта управления KUKA smartPAD

- |   |                             |   |                             |
|---|-----------------------------|---|-----------------------------|
| 1 | Переключатель подтверждения | 4 | Разъем USB                  |
| 2 | Клавиша пуска (зеленая)     | 5 | Переключатель подтверждения |
| 3 | Переключатель подтверждения | 6 | Заводская табличка          |

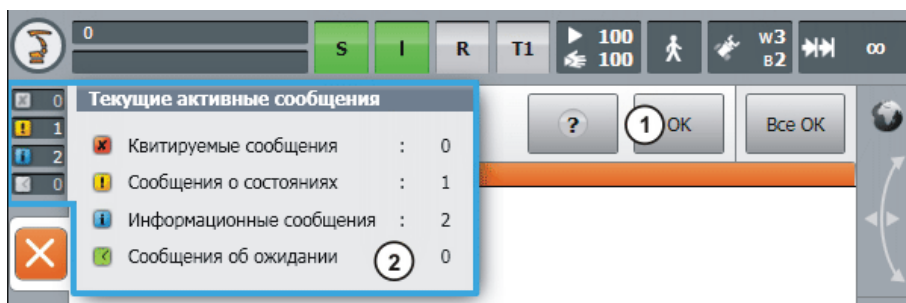
**Описание**

Элемент	Описание
<b>Заводская табличка</b>	Заводская табличка
<b>Клавиша пуска</b>	Клавишей пуска запускают программу.
<b>Переключатель подтверждения</b>	<p>Переключатель подтверждения имеет 3 положения:</p> <ul style="list-style-type: none"> <li>■ не нажат;</li> <li>■ среднее положение;</li> <li>■ нажат до отказа.</li> </ul> <p>В режимах T1 и T2 переключатель подтверждения должен удерживаться в среднем положении, чтобы манипулятор мог перемещаться.</p> <p>В обычном автоматическом и внешнем автоматическом режимах работы переключатель подтверждения не работает.</p>
<b>Разъем USB</b>	<p>Разъем USB можно использовать для резервного копирования и восстановления информации.</p> <p>Он предназначен только для USB-накопителей с файловой системой FAT32.</p>

**6.3 Чтение и интерпретация сообщений системы управления роботом**

Продолжительность: 00:30:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

**Обзор сообщений**







**Рис. 6-4: Окно сообщений и счетчик сообщений**


- 1 Окно сообщений: показано текущее сообщение
- 2 Счетчик сообщений: количество сообщений для каждого типа сообщений


С помощью окна сообщений система управления взаимодействует с оператором. Она поддерживает пять различных видов сообщений:

#### Обзор типов сообщений:

Символ	Тип
	<p><b>Квитируемое сообщение</b></p> <ul style="list-style-type: none"> <li>Предназначены для представления состояний, при которых для дальнейшей обработки программы робота требуется подтверждение оператора (напр., квитирование АВАРИЙНОГО ОСТАНОВА).</li> <li>Квитируемые сообщения всегда приводят к тому, что робот останавливается или не запускается.</li> </ul>
	<p><b>Сообщение о состоянии</b></p> <ul style="list-style-type: none"> <li>Сообщения о состоянии показывают текущее состояние системы управления (напр., «АВАРИЙНЫЙ ОСТАНОВ»)</li> <li>Сообщения о состоянии невозможно квитировать, пока это состояние сохраняется.</li> </ul>
	<p><b>Информационное сообщение</b></p> <ul style="list-style-type: none"> <li>Информационные сообщения предоставляют информацию для надлежащего управления роботом (напр., «Требуется нажатие кнопки запуска»).</li> <li>Информационные сообщения можно квитировать. Однако их квитирование необязательно, поскольку они не останавливают работу системы управления.</li> </ul>
	<p><b>Сообщение об ожидании</b></p> <ul style="list-style-type: none"> <li>Сообщения об ожидании указывают на событие (состояние, сигнал или время), которое ожидает система управления.</li> <li>Сообщения об ожидании можно вручную прервать нажатием на экранную кнопку «Выполнить моделирование».</li> </ul>

**УВЕДОМЛЕНИЕ** Команду «Выполнить моделирование» можно использовать только в том случае, если столкновения или другие опасности исключены!

	<p><b>Диалоговое сообщение</b></p> <ul style="list-style-type: none"> <li>Диалоговые сообщения используются для прямой связи с пользователем/запроса действия пользователя.</li> <li>Показывается окно сообщений с экранными кнопками, которые содержат различные варианты ответа.</li> </ul>
---	---

 Нажатием на кнопку **ОК** можно квитировать квитируемое сообщение. Нажатием на кнопку **Все ОК** можно сразу квитировать все квитируемые сообщения.

#### Влияние сообщений

Сообщения оказывают влияние на функциональность робота. Квитируемые сообщения всегда приводят к тому, что робот останавливается или не запускается. Чтобы робот начал выполнять перемещения, сначала следует квитировать сообщение.

Команда **ОК** (квитируание) требует от пользователя ознакомиться с текстом сообщения.

**Советы по обращению с сообщениями:**

- внимательно прочитать;
- сначала прочитать старые сообщения. Новое сообщение может быть следствием старого;
- не нажимать просто на кнопку «Все ОК»;
- особенно после запуска: просмотреть сообщения. Для этого вывести на экран все сообщения. Нажатие на окно сообщений раскроет список сообщений.

### Обращение с сообщениями

Сообщения всегда показываются с датой и временем, чтобы можно было воспроизвести точное время события.



Рис. 6-5: Квитирование сообщений

#### Порядок действий для просмотра и квитирования сообщений:

1. Коснуться окна сообщений, чтобы раскрыть список сообщений.
2. Квитирование:
  - квитировать отдельные сообщения нажатием на кнопку **ОК**;
  - альтернатива: квитировать все сообщения нажатием на кнопку **Все ОК**.
3. Повторное касание верхнего сообщения или касание значка «X» в левой части экрана снова закроет список сообщений.

## 6.4 Выбор и установка режима работы

Продолжительность: 00:15:00  
 Оснащение:  
 Информация:  
 Сертификат:

### Режимы работы робота KUKA

- T1 (вручную, пониженная скорость):
  - для режима тестирования, программирования и обучения;
  - скорость в программном режиме: макс. 250 мм/с;
  - скорость в ручном режиме: макс. 250 мм/с.
- T2 (вручную, высокая скорость):
  - для режима тестирования;
  - скорость в программном режиме соответственно запрограммированной скорости;
  - ручной режим: невозможен.
- AUT (автоматика):
  - для промышленных роботов без системы управления верхнего уровня;
  - скорость в программном режиме соответственно запрограммированной скорости;
  - ручной режим: невозможен.



- AUT EXT (внешняя автоматика):
  - для промышленных роботов с системой управления верхнего уровня (ПЛК);
  - скорость в программном режиме соответственно запрограммированной скорости;
  - ручной режим: невозможен.

### Указания по технике безопасности для режимов работы

#### Ручные режимы работы T1 и T2

Ручной режим работы предназначен для работ по наладке. Работы по наладке – это все работы, которые должны быть проведены в робототехнической системе, чтобы стала возможной ее эксплуатация в автоматическом режиме. К ним относятся:

- обучение/программирование;
- выполнение программ в пошаговом режиме (тестирование/проверка).

Новые или измененные программы сначала всегда должны быть испытаны в режиме работы **Вручную, пониженная скорость (T1)**.

Режим работы – **Вручную, пониженная скорость (T1)**:

- система защиты оператора (защитная дверца) неактивна;
- внутри пространства, ограниченного защитными приспособлениями, по возможности должны отсутствовать люди.  
Если все же необходимо, чтобы внутри пространства, ограниченного защитными приспособлениями, находилось несколько человек, учитывать следующие положения:
  - все люди должны иметь открытый вид на робототехническую систему;
  - все люди должны постоянно поддерживать между собой визуальный контакт;
- оператор должен занять положение, которое позволяет просматривать опасные зоны и тем самым предотвратить опасность.

Режим работы – **Вручную, высокая скорость (T2)**:

- система защиты оператора (защитная дверца) неактивна;
- данный режим работы разрешается использовать только в случае, если это требуется для проведения проверки с большей скоростью, чем в режиме «Вручную, пониженная скорость»;
- обучение в этом режиме работы не допускается;
- перед началом проверки оператор должен убедиться, что устройства подтверждения функционируют;
- оператор должен занять положение вне опасной зоны;
- внутри пространства, ограниченного защитными приспособлениями, должны отсутствовать люди.

Режимы работы **Автоматика** и **Внешняя автоматика**:

- предохранительные устройства и защитные приспособления должны иметься в наличии и полностью функционировать;
- все люди должны находиться вне пространства, ограниченного защитными приспособлениями.

### Порядок действий



При переключении режима работы во время эксплуатации происходит незамедлительное отключение приводов. Промышленный робот останавливается при безопасном останове 2.

1. На пульте КСР установить переключатель для вызова менеджера соединений в другое положение. На экране появится менеджер соединений.



2. Выбрать режим работы.



3. Установить переключатель для вызова менеджера соединений в исходное положение.

Выбранный режим работы появится в строке состояния пульта smartPAD.



## 6.5 Перемещение осей робота по отдельности

---

Продолжительность: 00:20:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

---

Описание:  
перемещение по  
осям

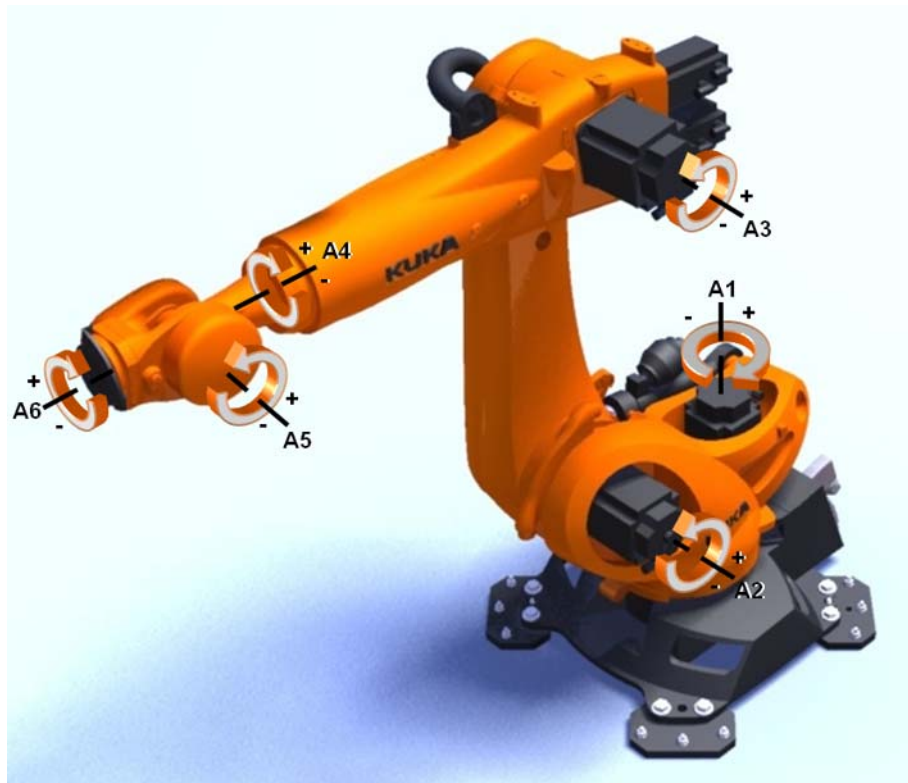


Рис. 6-6: Степени свободы робота KUKA

#### Перемещение осей робота

- Переместить каждую ось по отдельности в положительном или отрицательном направлении.
- Для этого используются клавиши перемещения или пространственная мышь KUKA smartPAD.
- Скорость можно изменить (ручная коррекция: NOV).
- Перемещение вручную возможно только в режиме работы T1.
- Кнопка подтверждения должна быть нажата.

#### Принцип

При объяснении принципа для перемещения робота следует показать процедуру на пульте smartPAD.

Приводы активируются нажатием на клавишу подтверждения. При нажатии клавиши перемещения или пространственной мыши запускается система регулирования осей робота и выполняется нужное перемещение.

Возможно как постоянное перемещение, так и инкрементальное перемещение. Для этого выбрать инкрементальный размер в строке состояния.

**На ручной режим работы оказывают действие следующие сообщения:**

Сообщение	Причина	Способ устранения
«Активные команды заблокированы»	Возникло сообщение или состояние (останова), которое привело к блокированию активных команд (напр., нажата кнопка аварийного останова или приводы еще не готовы).	Деблокировать кнопку аварийного останова и/или квитировать сообщения в окне сообщений. После нажатия клавиши подтверждения приводы немедленно станут доступны.
«Программный конечный выключатель A5»	Программный конечный выключатель показанной оси (напр., A5) был подведен в указанном направлении («+» или «-»).	Переместить указанную ось в противоположном направлении.

**Указания по технике безопасности для перемещения по осям вручную**

**Режим работы**

Ручное управление роботом разрешено только в режиме работы T1 (вручную, пониженная скорость). Скорость перемещения вручную в режиме T1 составляет макс. 250 мм/с. Режим работы устанавливается с помощью менеджера соединений.

**Кнопка подтверждения**

Чтобы перемещать робот, кнопка подтверждения должна быть нажата. На пульте smartPAD размещено три кнопки подтверждения. Каждая из кнопок подтверждения имеет три положения:

- не нажата;
- среднее положение;
- нажата до отказа (экстренное положение).

**Программный конечный выключатель**

Перемещение робота при перемещении по осям вручную также ограничено положительным и отрицательным максимальными значениями программного конечного выключателя.

**УВЕДОМЛЕНИЕ**

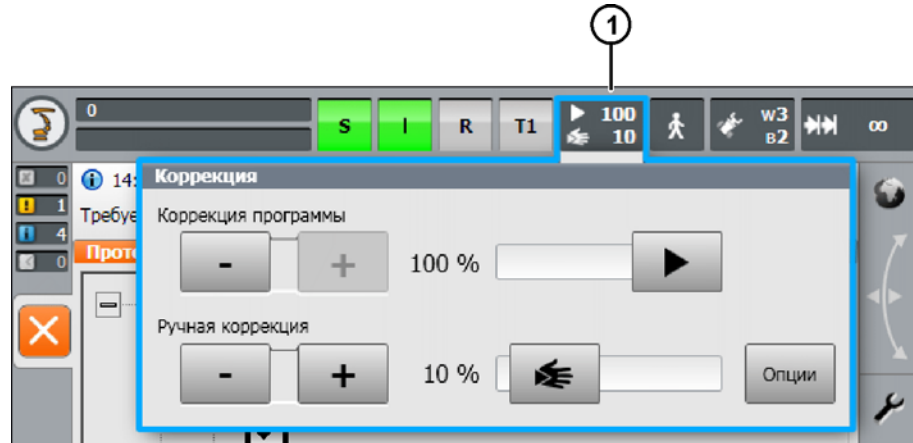
Если в окне сообщений появится сообщение «Выполнить юстировку», возможно перемещение за эти пределы. Это может привести к повреждению системы робота.

**Порядок действий: выполнение осевого перемещения**

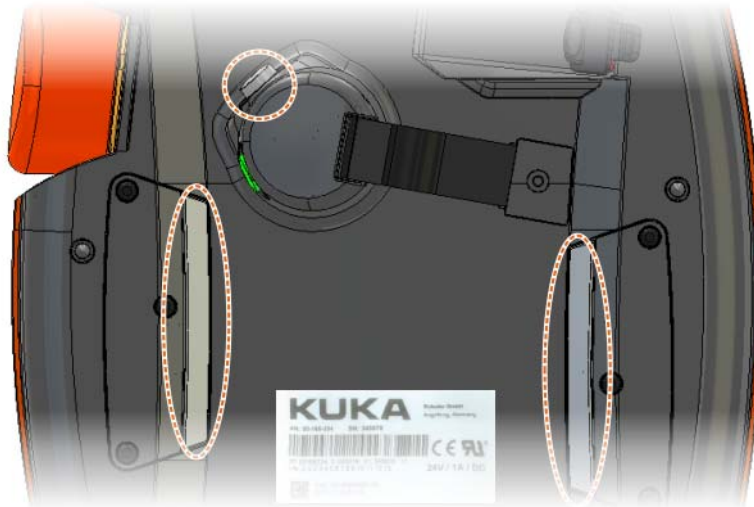
1. Выбрать в качестве опции для клавиш перемещения **Оси**.



- Установить ручную коррекцию.



- Установить переключатель подтверждения в среднее положение и удерживать переключатель в нем.



Рядом с клавишами перемещения показаны оси от A1 до A6.

- Нажать клавишу перемещения «плюс» или «минус» для перемещения по оси в положительном или отрицательном направлении.



**Перемещение  
робота в  
аварийной  
ситуации без  
системы управ-  
ления**



**Рис. 6-7: Устройство свободного вращения**



**Описание**

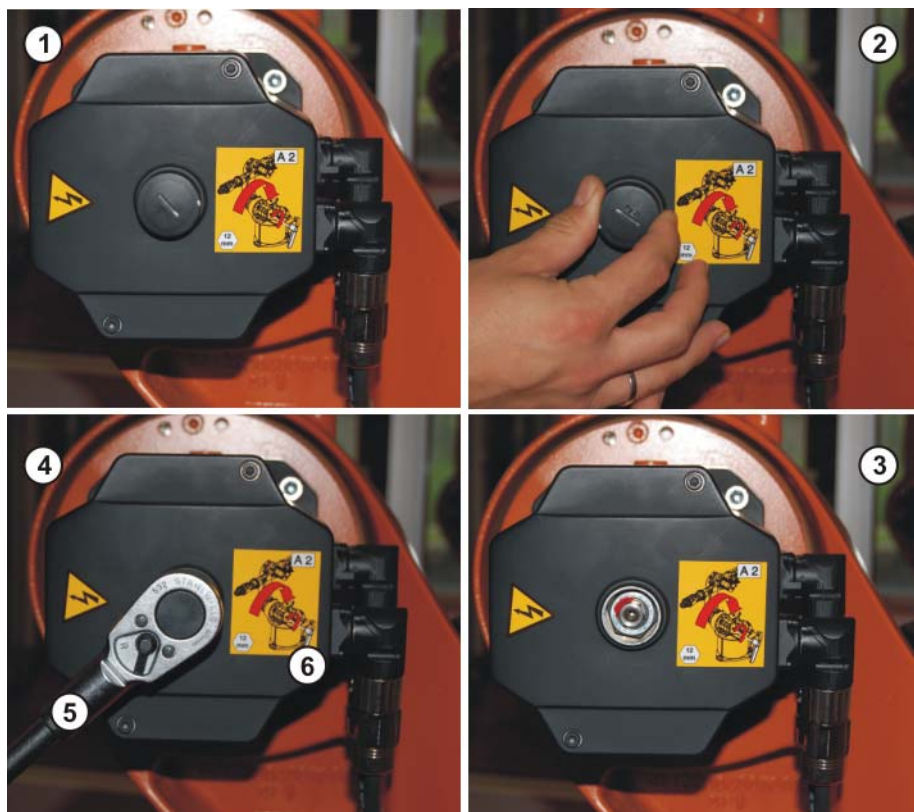
В аварийной ситуации или в случае возникновения неисправности перемещение робота может быть осуществлено механически при помощи устройства свободного вращения. Устройство свободного вращения также может быть использовано для приводных двигателей основной оси и для приводных двигателей ручной оси в зависимости от исполнения робота. Данное устройство разрешается применять **только** в исключительных случаях и в аварийных ситуациях, например, для освобождения людей. После использования устройства свободного вращения при необходимости следует заменить соответствующие двигатели.

**⚠ ОСТОРОЖНО** Во время эксплуатации двигателя достигают высоких температур, в связи с чем возникает опасность получения ожогов кожи. Не касаться двигателей! Принять соответствующие меры безопасности, например, надеть защитные перчатки.

**Порядок действий**

1. выключить систему управления роботом и предохранить ее от несанкционированного включения (например, с помощью висячего замка);
2. Снять с двигателя защитную крышку.
3. Установить устройство свободного вращения на соответствующий двигатель и переместить ось в требуемом направлении.

В качестве опции можно заказать маркировку направлений со стрелками на двигателях. Сопrotивление механического тормоза двигателя и дополнительные осевые нагрузки должны быть преодолены.



**Рис. 6-8: Порядок действий с устройством свободного вращения**

Поз.	Описание
1	Двигатель A2 с закрытой защитной крышкой
2	Открывание защитной крышки на двигателе A2
3	Двигатель A2 со снятой защитной крышкой

Поз.	Описание
4	Установка устройства свободного вращения на двигатель A2
5	Устройство свободного вращения
6	Табличка (опция) с описанием направления вращения

**⚠ ВНИМАНИЕ** При перемещении оси посредством устройства свободного вращения существует риск повреждения тормоза двигателя. Это может привести к травмам персонала и материальному ущербу. По окончании использования устройства свободного вращения необходимо заменить соответствующий двигатель.

**i** Дополнительные сведения приведены в инструкции по эксплуатации и монтажу робота.

## 6.6 Системы координат относительно робота

---

Продолжительность: 00:20:00  
 Оснащение: Пересечение координат  
 Информация:  
 Сертификат:

---

При управлении, программировании и вводе промышленных роботов в эксплуатацию системы координат имеют большое значение. В системе управления роботом заданы следующие системы координат:

- WORLD: универсальная система координат;
- ROBROOT: система координат основания робота;
- BASE: основная система координат;
- FLANGE: система координат фланца;
- TOOL: система координат инструмента.



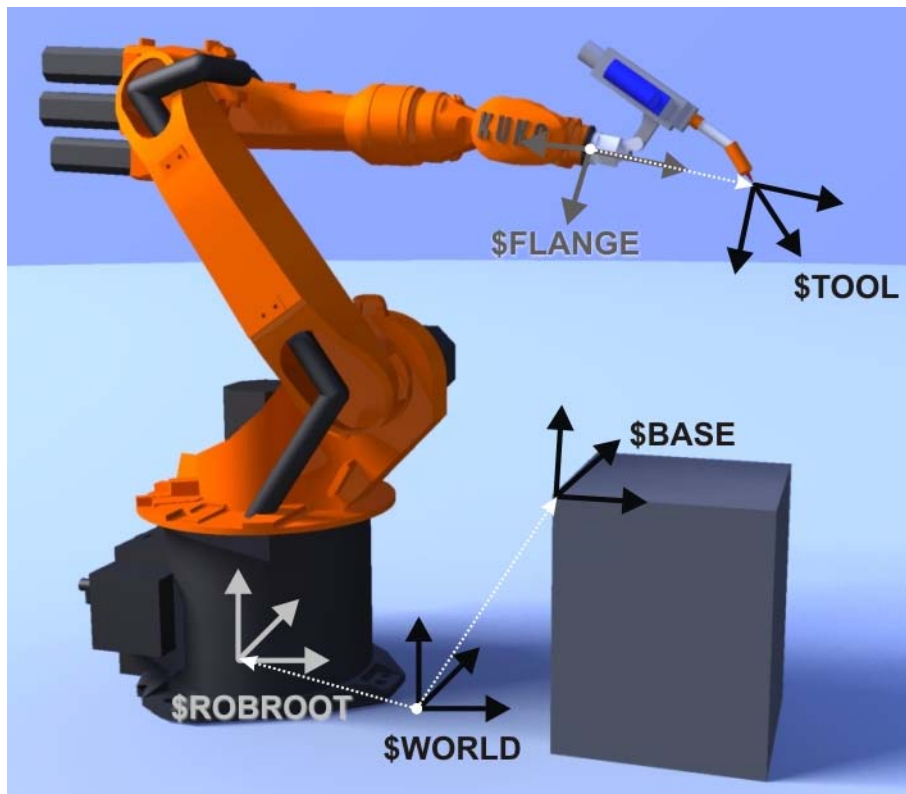


Рис. 6-9: Системы координат на роботе KUKA

Имя	Место	Использование	Особенность
<b>WORLD</b>	Свободно определяемая	Начало координат для ROBROOT и BASE	В большинстве случаев находится в основании робота
<b>ROBROOT</b>	Всегда в основании робота	Начало координат робота	Описывает положение робота по отношению к системе координат WORLD
<b>BASE</b>	Свободно определяемая	Заготовки, приспособления	Описывает положение базы по отношению к системе координат WORLD
<b>FLANGE</b>	Всегда на фланце робота	Начало координат для системы координат TOOL	Начало координат – середина фланца робота
<b>TOOL</b>	Свободно определяемая	Инструменты	Начало системы координат TOOL обозначается как <b>TCP</b>  (TCP = центральная точка инструмента)

## 6.7 Перемещение робота в универсальной системе координат

---

Продолжительность:	00:20:00
Оснащение:	Офисный ПК
Информация:	
Сертификат:	

---

### Перемещение в универсальной системе координат

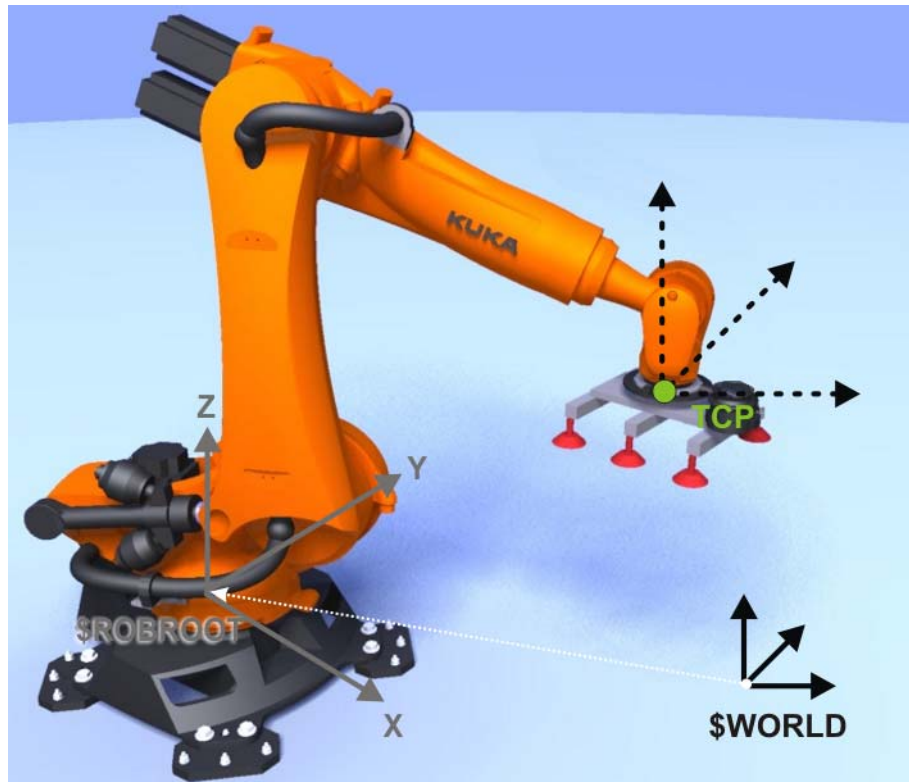


Рис. 6-10: Принцип перемещения вручную в универсальной системе координат

- Инструмент робота может перемещаться согласно направлениям координат универсальной системы координат. При этом перемещаются **все** оси робота.
- Для этого используются клавиши перемещения или пространственная мышь KUKA smartPAD.
- В стандартной установке универсальная система координат находится в основании робота.
- Скорость можно изменить (ручная коррекция: HOV).
- Перемещение вручную возможно только в режиме работы T1.
- Кнопка подтверждения должна быть нажата.

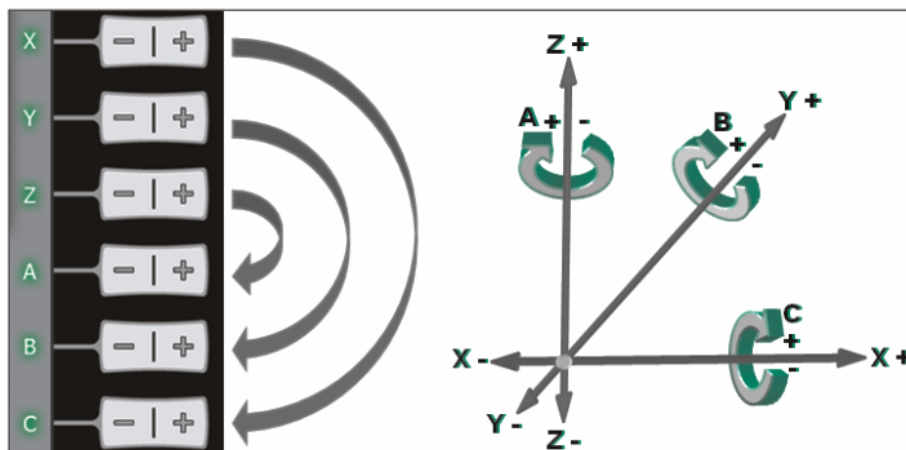
#### Пространственная мышь

- Пространственная мышь обеспечивает интуитивное перемещение робота и является идеальным решением для перемещения вручную в универсальной системе координат.
- Положение мыши и степени свободы можно изменять.

**Принцип перемещения вручную в универсальной системе координат**

**Робот может перемещаться в системе координат двумя различными способами:**

- линейно (прямолинейно) вдоль направлений ориентации системы координат: X, Y, Z;
- вращательно (поворачиваясь/наклоняясь) вокруг направлений ориентации системы координат: угол A, B и C.



**Рис. 6-11: Система прямоугольных координат**

При команде перемещения (напр., нажатии на клавишу перемещения) система управления сначала рассчитывает отрезок. Исходной точкой отрезка является отсчетная точка инструмента (TCP). Направление отрезка устанавливает универсальная система координат. Система управления управляет всеми осями таким образом, что инструмент проводится по этому отрезку (перенос) или вращается вокруг него (поворот).

**Преимущества использования универсальной системы координат:**

- перемещение робота всегда является предсказуемым;
- перемещения всегда однозначны, поскольку начало и направления координат всегда известны;
- универсальная система координат всегда доступна для применения на юстированном роботе;
- пространственная мышь обеспечивает интуитивное управление.

**Использование пространственной мыши**

- С помощью пространственной мыши возможны все виды перемещений:
  - линейно: посредством нажатия и вытягивания пространственной мыши;

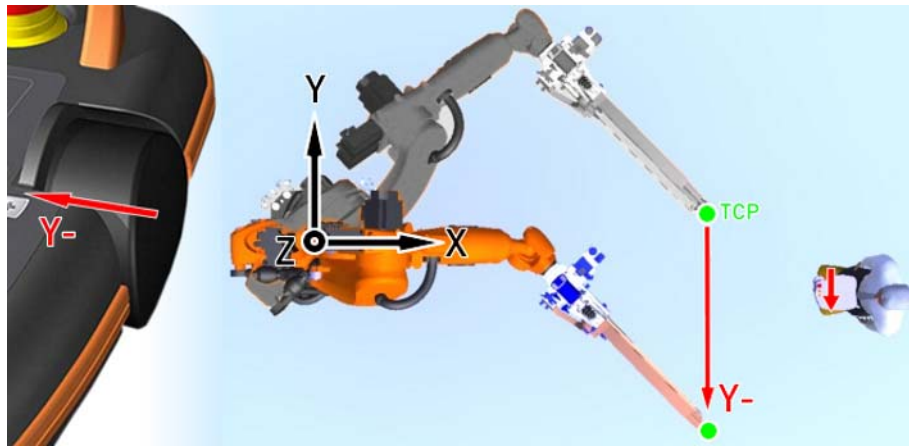


Рис. 6-12: Пример: перемещение влево

- вращательно: посредством поворота и наклона пространственной мыши;

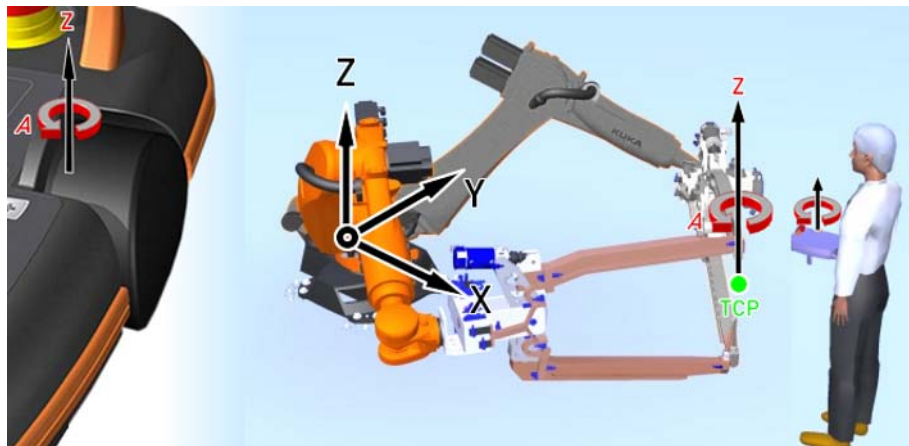


Рис. 6-13: Пример: вращательное перемещение вокруг оси Z: угол A

- соответственно положению человека и робота можно отрегулировать положение пространственной мыши.

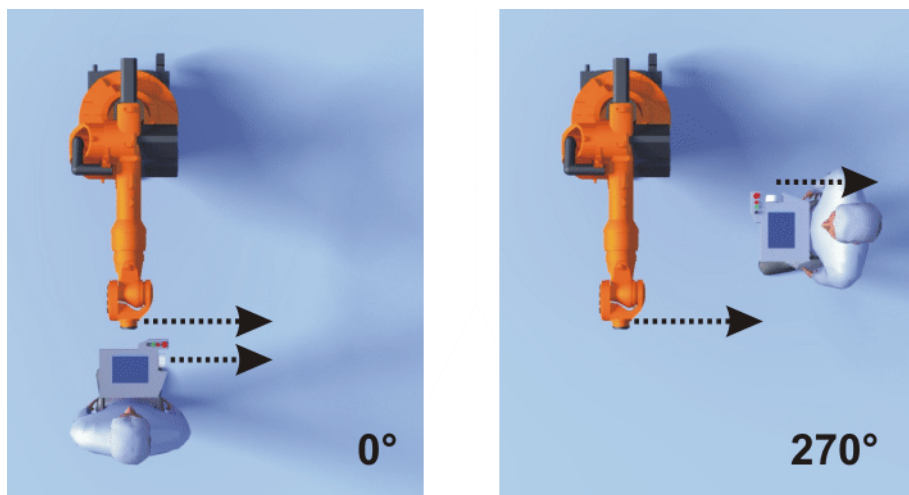
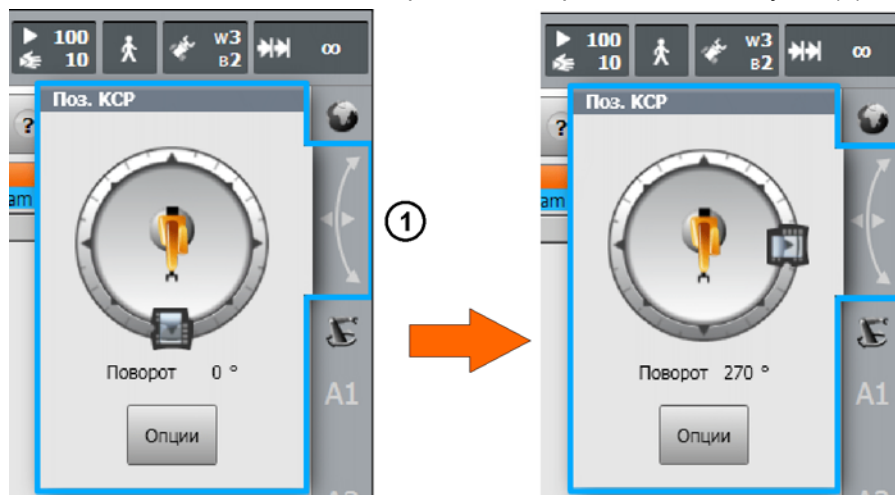


Рис. 6-14: Пространственная мышь: 0° и 270°

**Выполнение  
линейных  
перемещений  
(унив.)**

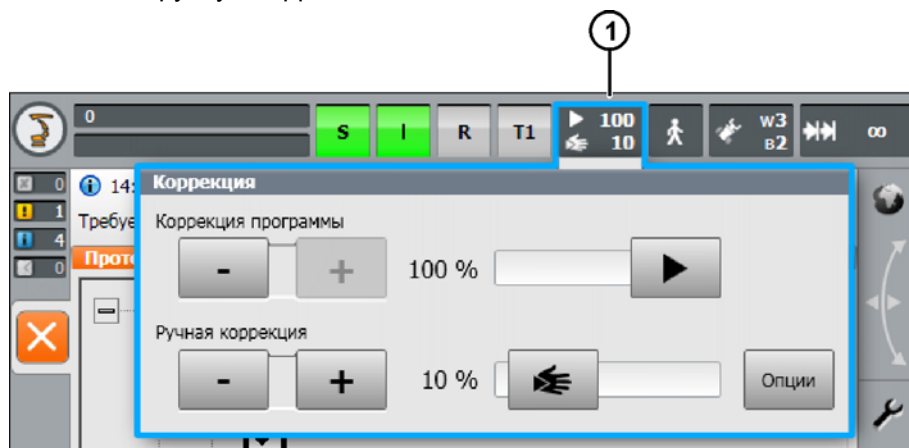
1. Установить положение КСР посредством перемещения ползунка (1).



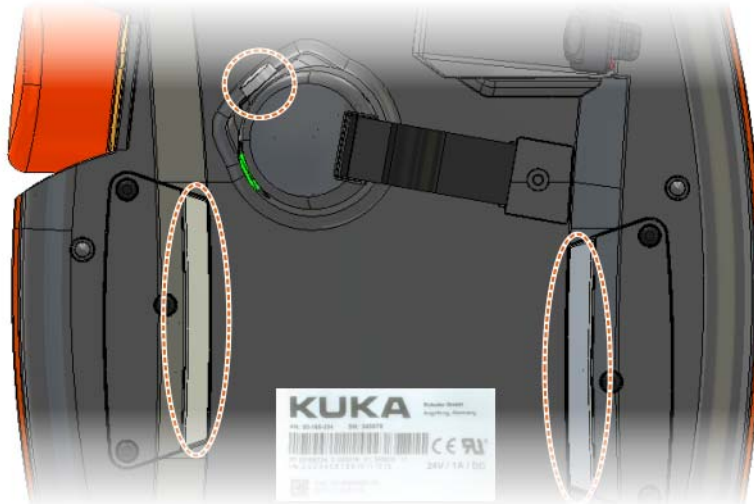
2. Выбрать в качестве опции для пространственной мыши **Унив..**



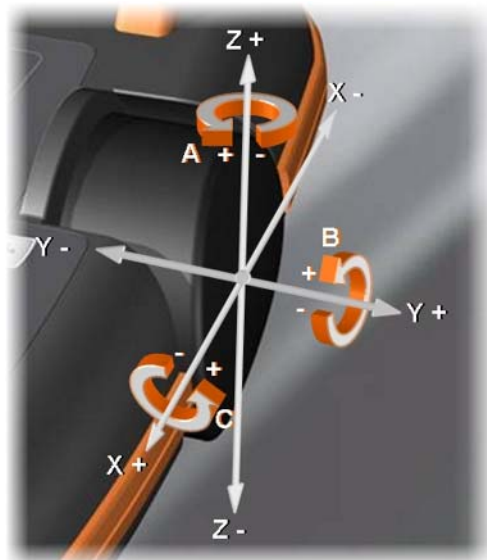
3. Установить ручную коррекцию.



- Установить переключатель подтверждения в среднее положение и удерживать переключатель в нем.



- Переместить робот с помощью пространственной мыши в соответствующем направлении.



- Также в качестве альтернативы можно использовать клавиши перемещения.





## 6.8 Перемещение робота в системе координат инструмента

Продолжительность: 00:20:00  
 Оснащение:  
 Информация:  
 Сертификат:

Перемещение вручную в системе координат инструмента

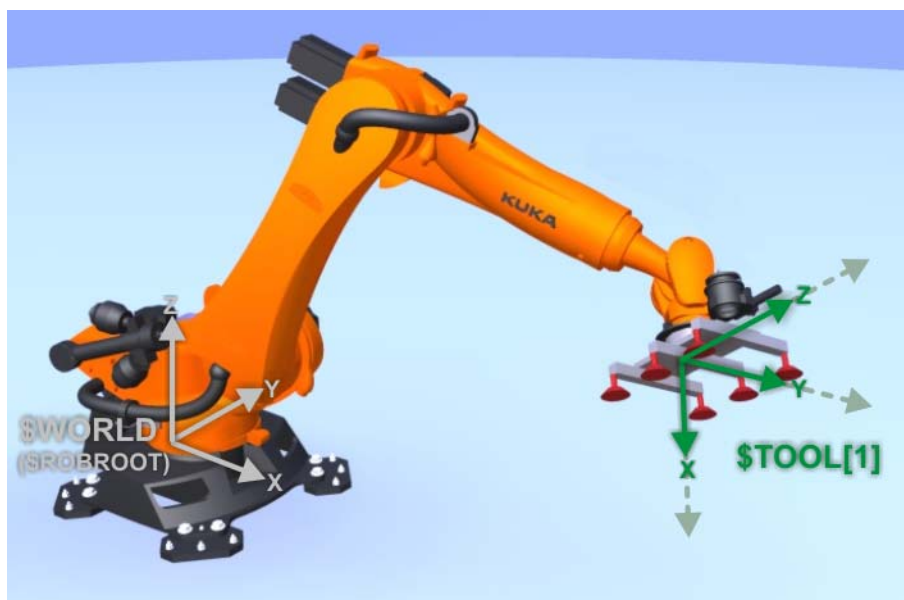


Рис. 6-15: Система координат Tool робота

- При перемещении вручную в системе координат инструмента можно перемещать робот соответственно направлениям координат заранее откалиброванного инструмента.

Таким образом, система координат не является неподвижной (ср. с универсальной/основной системой координат), а ведется роботом.

При этом перемещаются **все** требуемые оси робота. Выбор осей определяется системой и зависит от перемещения.

Начало системы координат инструмента называется **TCP** и соответствует рабочей точке инструмента.

- Для этого используются клавиши перемещения или пространственная мышь пульта KUKA smartPAD.
- Для выбора доступны 16 различных систем координат инструмента.
- Скорость можно изменить (ручная коррекция: HOV).
- Перемещение вручную возможно только в режиме работы T1.
- Кнопка подтверждения должна быть нажата.





Принцип перемещения вручную в системе координат инструмента

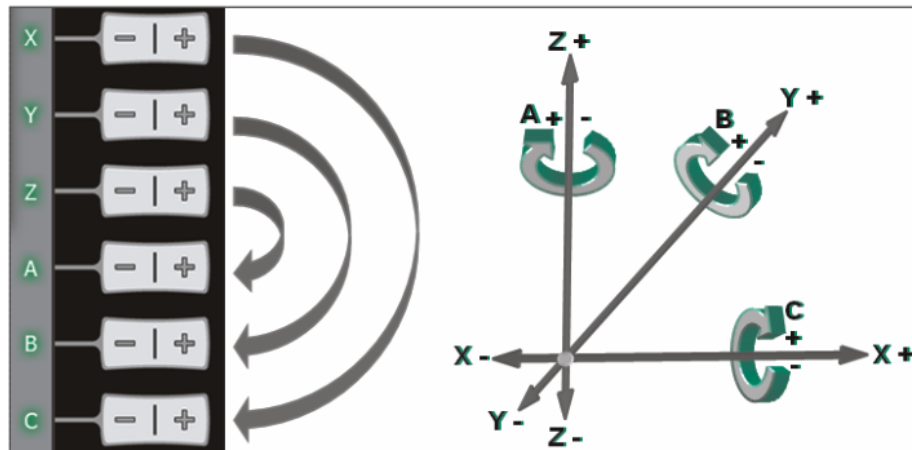


Рис. 6-16: Система прямоугольных координат

Робот может перемещаться в системе координат двумя различными способами:

- линейно (прямолинейно) вдоль направлений ориентации системы координат: X, Y, Z:
- вращательно (поворачиваясь/наклоняясь) вокруг направлений ориентации системы координат: угол A, B и C.

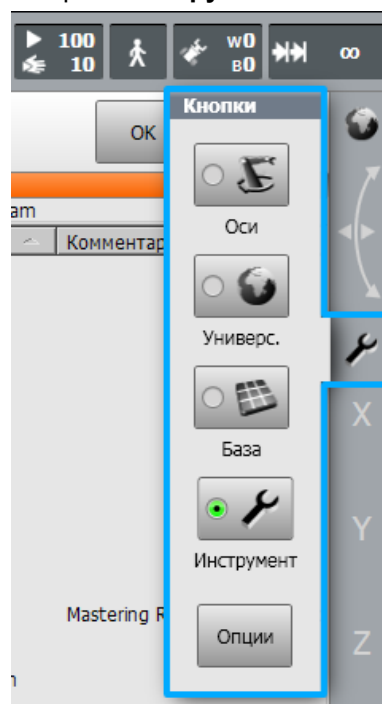
**Преимущества использования системы координат инструмента:**

- перемещение робота всегда является предсказуемым, пока известна система координат инструмента;
- возможно перемещение в направлении удара инструмента или для ориентирования точки TCP.

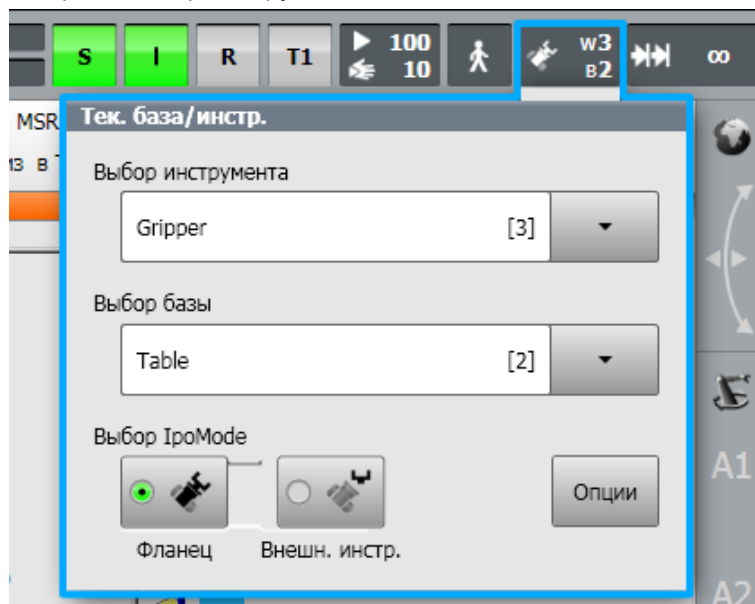
Под *направлением удара инструмента* понимают направление работы или действия инструмента: направление выпуска клея из сопла для нанесения клея, направление захвата при захвате детали и т. д.

Порядок действий

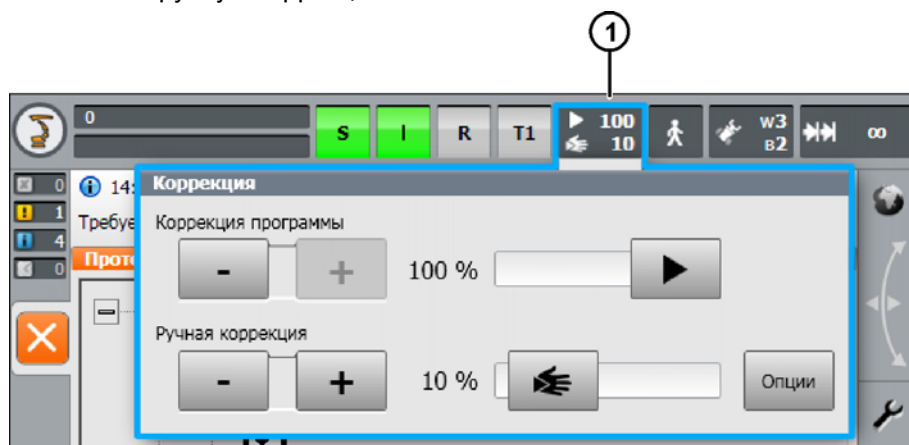
1. Выбрать **Инструмент** в качестве используемой системы координат.



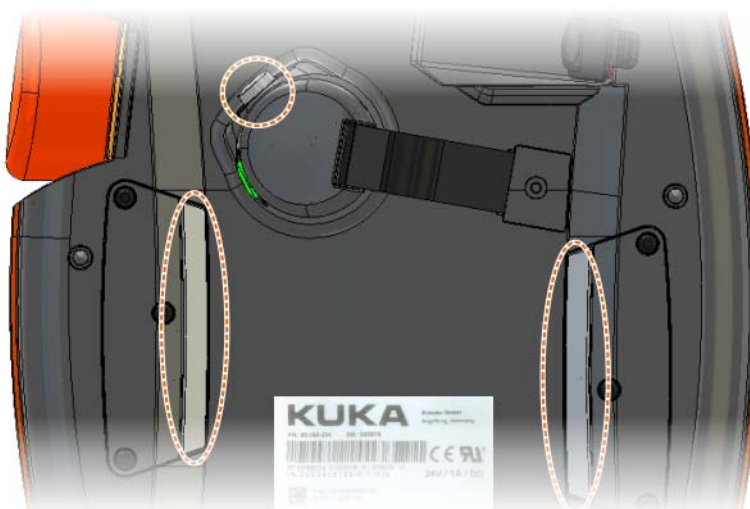
2. Выбрать номер инструмента.



3. Установить ручную коррекцию.



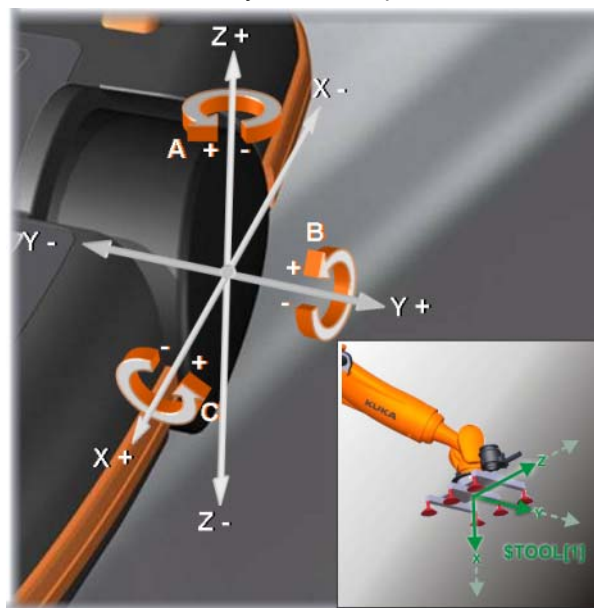
4. Установить переключатель подтверждения в среднее положение и удерживать переключатель в нем.



5. Переместить робот с помощью клавиш перемещения.



6. Альтернатива: переместить робот с помощью пространственной мыши в соответствующем направлении.



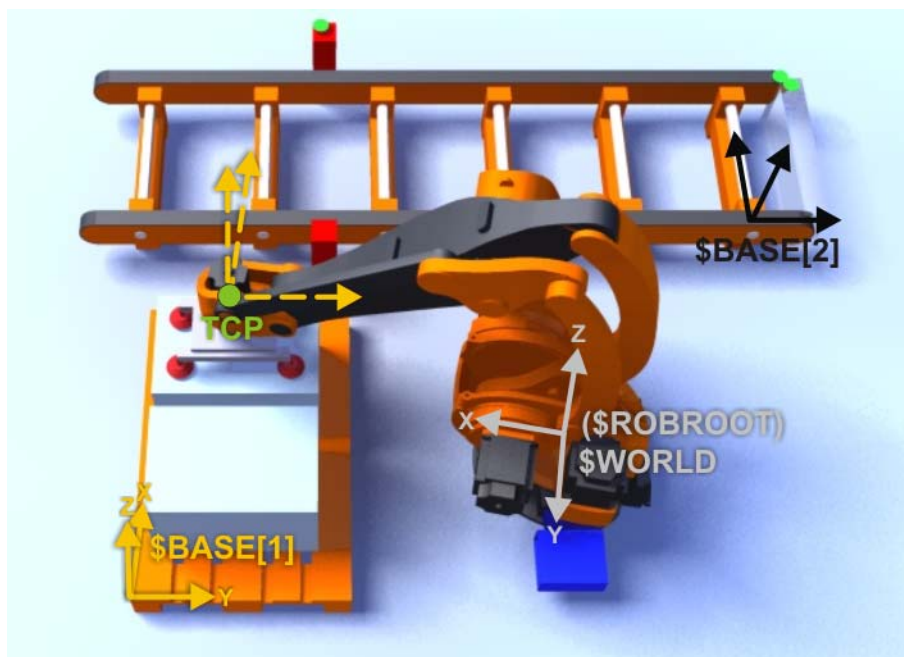
## 6.9 Перемещение робота в основной системе координат

---

Продолжи-  
тельность: 00:20:00  
Оснащение:  
Информация:  
Сертификат:

---

**Перемещение в основной системе координат**

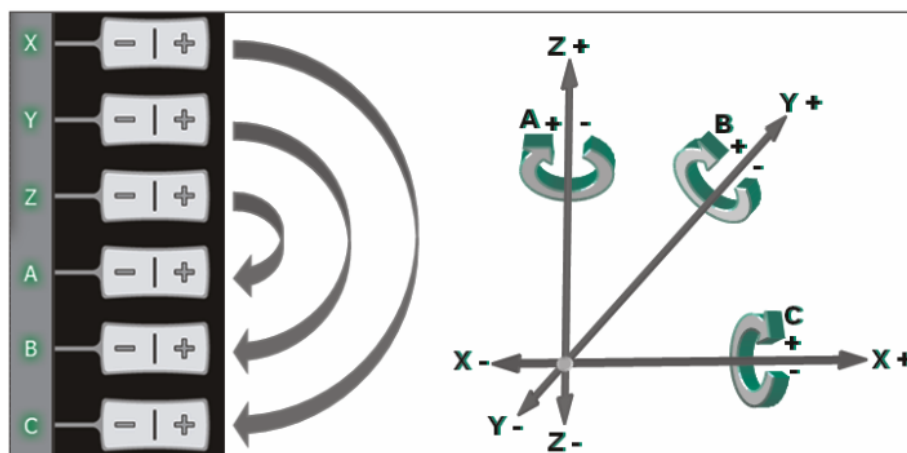


**Рис. 6-17: Перемещение вручную в основной системе координат**

**Описание базы**

- Инструмент робота может перемещаться согласно направлениям координат основной системы координат. Основные системы координат можно откалибровать индивидуально; зачастую они ориентированы вдоль кромок заготовок, захватов для заготовок или палет. Благодаря этому возможно удобное перемещение вручную.
- При этом перемещаются **все** требуемые оси робота. Выбор осей определяется системой и зависит от перемещения.
- Для этого используются клавиши перемещения или пространственная мышь KUKA smartPAD.
- Для выбора доступны 32 основные системы координат.
- Скорость можно изменить (ручная коррекция: HOV).
- Перемещение вручную возможно только в режиме работы T1.
- Кнопка подтверждения должна быть нажата.

**Принцип перемещения вручную в основной системе координат**



**Рис. 6-18: Система прямоугольных координат**


**Робот может перемещаться в системе координат двумя различными способами:**

- линейно (прямолинейно) вдоль направлений ориентации системы координат: X, Y, Z;
- вращательно (поворачиваясь/наклоняясь) вокруг направлений ориентации системы координат: угол A, B и C.

При команде перемещения (напр., нажатии на клавишу перемещения) система управления сначала рассчитывает отрезок. Исходной точкой отрезка является отсчетная точка инструмента (TCP). Направление отрезка устанавливает универсальная система координат. Система управления управляет всеми осями таким образом, что инструмент проводится по этому отрезку (перенос) или вращается вокруг него (поворот).

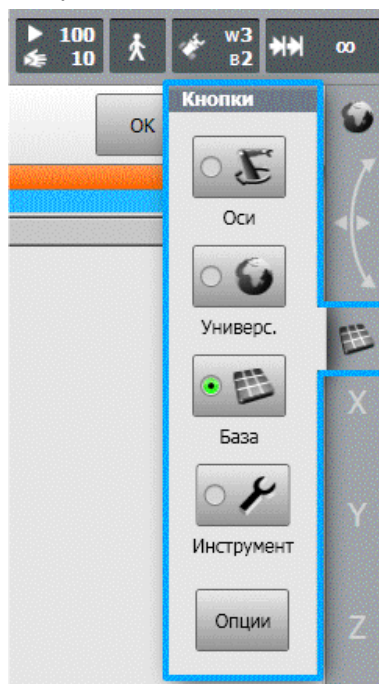
**Преимущества использования основной системы координат:**

- Перемещение робота всегда является предсказуемым, пока известна основная система координат.
- Также с помощью пространственной мыши возможно интуитивное управление. Условием является правильное положение оператора по отношению к роботу или основной системе координат.

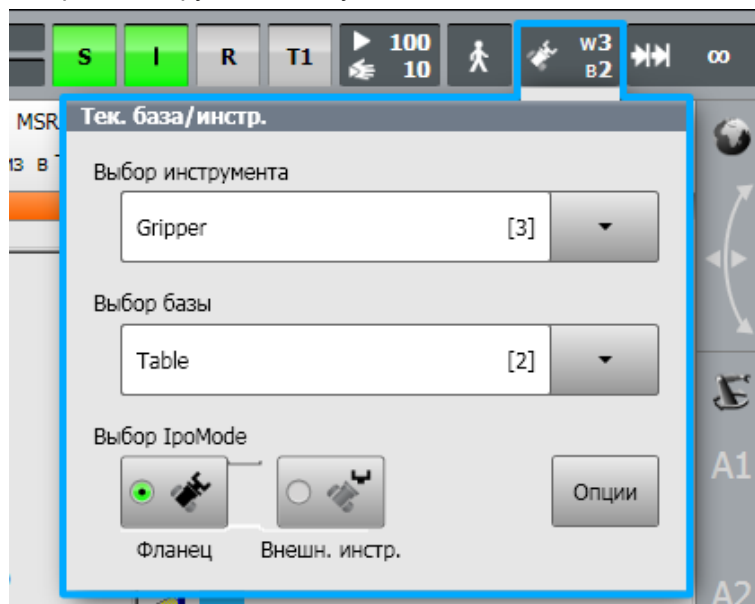
 Если также установлена верная система координат инструмента, в основной системе координат можно выполнить переориентацию вокруг точки TCP.

**Порядок действий**

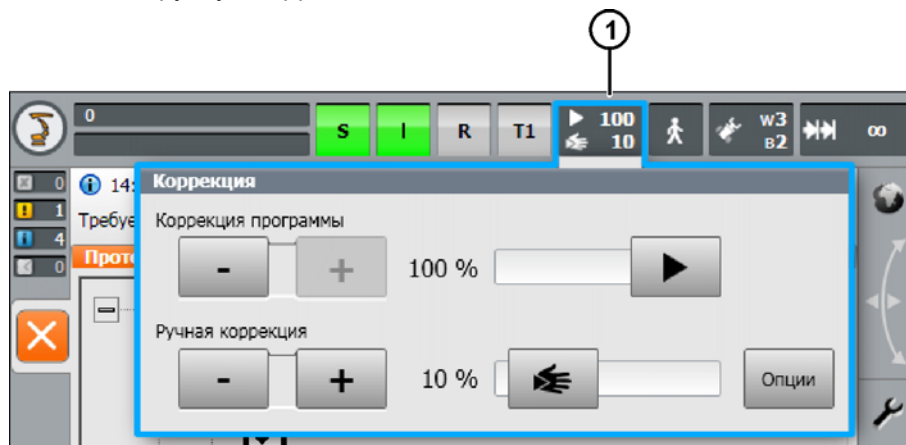
1. Выбрать в качестве опции для клавиш перемещения **База**.



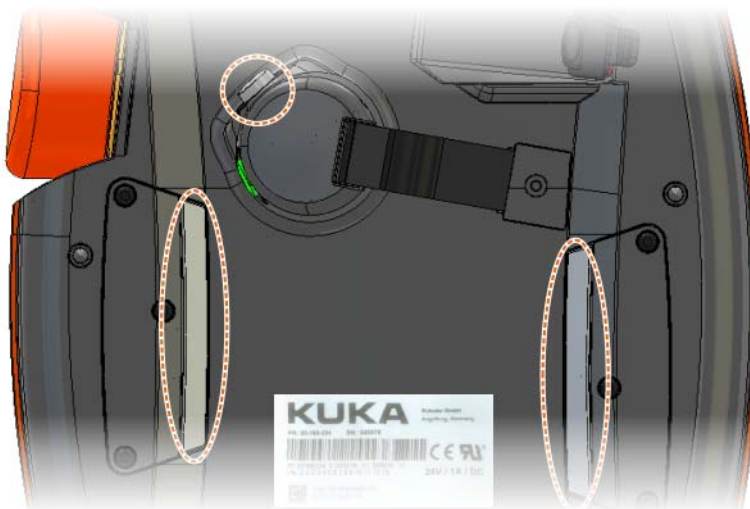
2. Выбрать инструмент и базу.



3. Установить ручную коррекцию.



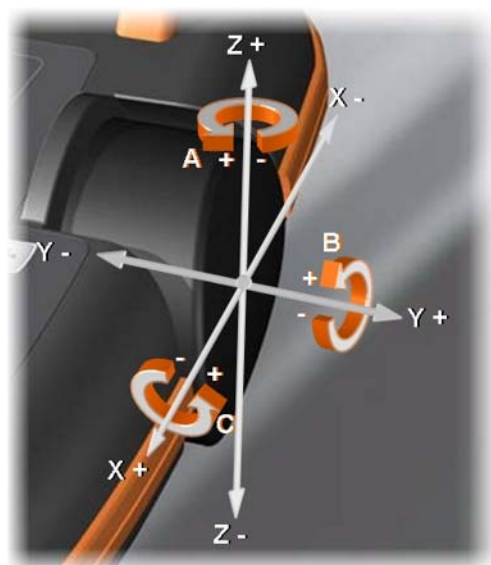
4. Установить переключатель подтверждения в среднее положение и удерживать переключатель в нем.



5. Выполнить перемещение с помощью клавиш перемещения в нужном направлении.



6. В качестве альтернативы можно выполнить перемещение с помощью пространственной мыши.







## 7 Ввод в эксплуатацию

### 7.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Юстировка робота
- Калибровка инструмента
- Данные нагрузки
- Калибровка базы
- Отображение актуальной позиции робота



Рис. 7-1: Смена главы

### 7.2 Принцип юстировки

#### Зачем нужна юстировка?

Промышленный робот можно оптимально использовать только в том случае, если он полностью и верно отъюстирован. Только в этом случае обеспечивается полная точность прохождения точек и траектории и вообще возможны запрограммированные перемещения.



При юстировке каждой оси робота присваивается отсчетное значение.

Полный процесс юстировки включает в себя юстировку каждой отдельной оси. С помощью технического вспомогательного средства (EMD = электронное наладочное устройство) каждой оси в **механическом нулевом положении** присваивается отсчетное значение (напр., 0°). Поскольку таким образом механические и электрические положения осей приводятся в соответствие, каждая ось получает однозначное значение угла.

Положение юстировки является сходным для всех роботов, но все же не одинаковым. Даже отдельные роботы одного типа могут иметь некоторые различия в точности положения.

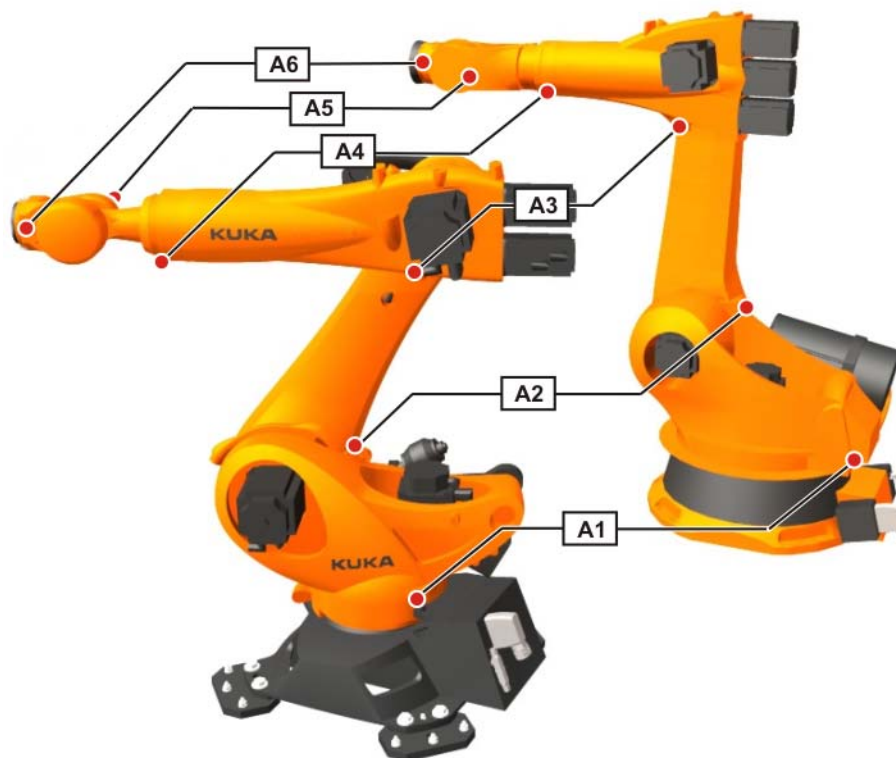


Рис. 7-2: Положение юстировочных патронов

Угловые значения механических нулевых положений (= отсчетные значения)

Ось	Поколение роботов Quantec	Другие типы роботов (напр., серия 2000, KR 16 и т. д.)
A1	-20°	0°
A2	-120°	-90°
A3	+110°	+90°
A4	0°	0°
A5	0°	0°
A6	0°	0°

**Когда выполняется юстировка?**

В целом, робот всегда должен быть отъюстирован. Юстировку следует выполнить в следующих случаях:

- при вводе в эксплуатацию;
- после проведения профилактических мероприятий на компонентах, которые задействованы в определении значения положения (напр., двигатель с синус.-косинус. преобразователем или RDC);
- если оси робота перемещались без системы управления, напр., с помощью устройства свободного вращения;
- после механических ремонтных работ/неисправностей сначала следует деюстировать робот перед тем, как можно будет провести юстировку:
  - после замены редуктора;
  - после перемещения до концевого упора со скоростью более 250 мм/с;
  - после столкновения.

**i** Перед проведением профилактических мероприятий целесообразно проверить текущую юстировку.

**ВНУТРЕННЯЯ ИНФОРМАЦИЯ:** подробнее объяснить проверку юстировки: юстировка заранее проверяется, чтобы определить наличие ошибок в самой юстировке. При необходимости после ремонта можно вручную выполнить подвод к значению неверной юстировки и с помощью временной юстировки повторно активировать это значение в качестве положения юстировки.

#### Указания по технике безопасности для юстировки

При неотъюстированных осях работа функции робота значительно ограничена:

- программный режим невозможен: невозможно обработать запрограммированные точки;
- линейное перемещение вручную невозможно: перемещения в системах координат невозможны;
- программные концевые выключатели деактивированы.

**УВЕДОМЛЕНИЕ** При деюстированном роботе программные концевые выключатели деактивированы. Робот может дойти до амортизаторов на концевых упорах, что может вызвать его повреждение, а амортизаторы придется заменить. Деюстированный робот по возможности не перемещать или в максимально возможной степени уменьшить ручную коррекцию.

#### Выполнение юстировки



**Рис. 7-3: Использование EMD**

В процессе юстировки определяется механическая нулевая точка оси. При этом ось перемещается до тех пор, пока не будет достигнута механическая нулевая точка. Это верно, когда измерительный штифт достиг самой нижней точки в измерительной насечке. Поэтому каждая ось оснащена юстировочным патроном и юстировочной меткой.

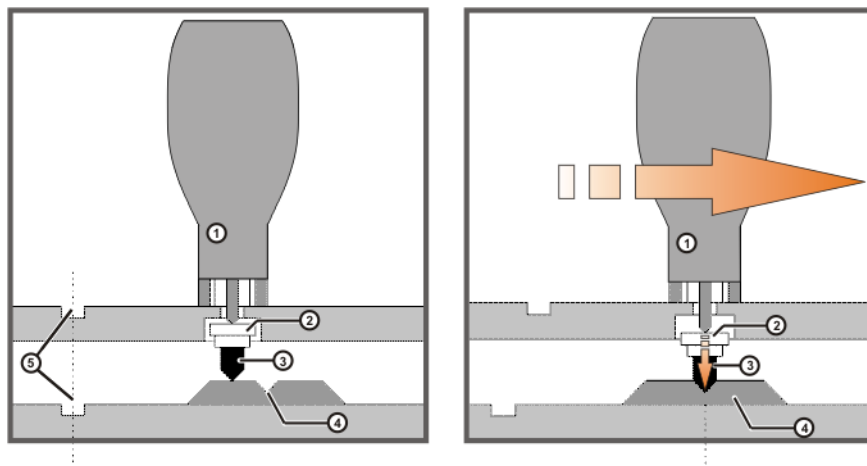


Рис. 7-4: Процесс юстировки EMD

- |   |   |   |                                       |
|---|---|---|---------------------------------------|
| 1 | Электронное наладочное устройство (EMD) | 4 | Измерительная насечка                 |
| 2 | Измерительный патрон                    | 5 | Предварительное обозначение юстировки |
| 3 | Измерительный штифт                     |   |                                       |

### 7.3 Юстировка робота

---

Продолжительность: 00:30:00  
 Оснащение:  
 Информация:  
 Сертификат:

---

## Возможности для юстировки робота



Рис. 7-5: Возможности юстировки

## Порядок действий при стандартной юстировке

1. Привести робот в предъюстировочное положение.



Рис. 7-6: Примеры предъюстировочного положения

2. В главном меню выбрать пункт **Ввод в эксплуатацию > Юстировка > EMD > Стандарт > Задать юстировку**.  
Открывается окно. На дисплее отображаются все юстируемые оси. Выделяется ось с самым низким номером.
3. Снять с оси, выделенной в окне, защитную крышку измерительного патрона. Повернутое устройство EMD может служить в качестве отвертки. Навинтить устройство EMD на измерительный патрон.



Рис. 7-7: Привинченное к измерительному патрону устройство EMD

4. После этого подсоединить измерительный провод к устройству EMD и подключить его к разъему X32 на коробке выводов робота.



Рис. 7-8: Провод EMD, подключенный

**⚠ ВНИМАНИЕ** Всегда привинчивать устройство EMD к измерительному патрону без измерительного провода. Только после этого подключить измерительный провод к устройству EMD. Иначе возможно повреждение измерительного провода. Точно так же при демонтаже устройства EMD сначала следует снять измерительный провод с устройства EMD. Только после этого снять устройство EMD с измерительного патрона. После выполнения юстировки отсоединить измерительный провод от разъема X32. В противном случае это может привести к помехам или материальному ущербу.

5. Нажать кнопку **Юстировка**.
6. Нажать и удерживать нажатым переключатель подтверждения в среднем положении; нажать и удерживать нажатой клавишу запуска.



Рис. 7-9: Клавиша подтверждения и запуска



Положение юстировки достигается, когда устройство EMD пройдет самую нижнюю точку измерительной насечки. Робот автоматически останавливается. Значения сохраняются в памяти. Ось в окне затемняется.

7. Отсоединить измерительный провод от устройства EMD. Затем снять устройство EMD с измерительного патрона и снова установить защитную крышку.
8. Повторить шаги 2 - 5 для всех юстируемых осей.
9. Закрыть окно.
10. Отсоединить измерительный провод от разъема X32.

**Для чего нужна установка/ проверка юстировки нагрузки со смещением**

Применение

- Проверка юстировки нагрузки используется для того, чтобы проверить первичную юстировку при установленном инструменте.
- Установка юстировки нагрузки со смещением используется для восстановления утерянной первичной юстировки. Так как заученное смещение сохраняется и при утрате юстировки, система управления роботом может рассчитать первичную юстировку.

Необходимые условия:

- условия соответствуют условиям первичной юстировки;
- на работе установлена нагрузка, для которой выполнено обучение смещения;
- все оси находятся в предъюстировочном положении;
- ни одна из программ не выбрана;
- выбран режим работы T1.

Допустимые значения допусков при проверке юстировки:

- ок. 1800 инкрементов.

## 7.4 Нагрузки на работе

Продолжительность:	00:05:00
Оснащение:	Модель робота
Информация:	
Сертификат:	

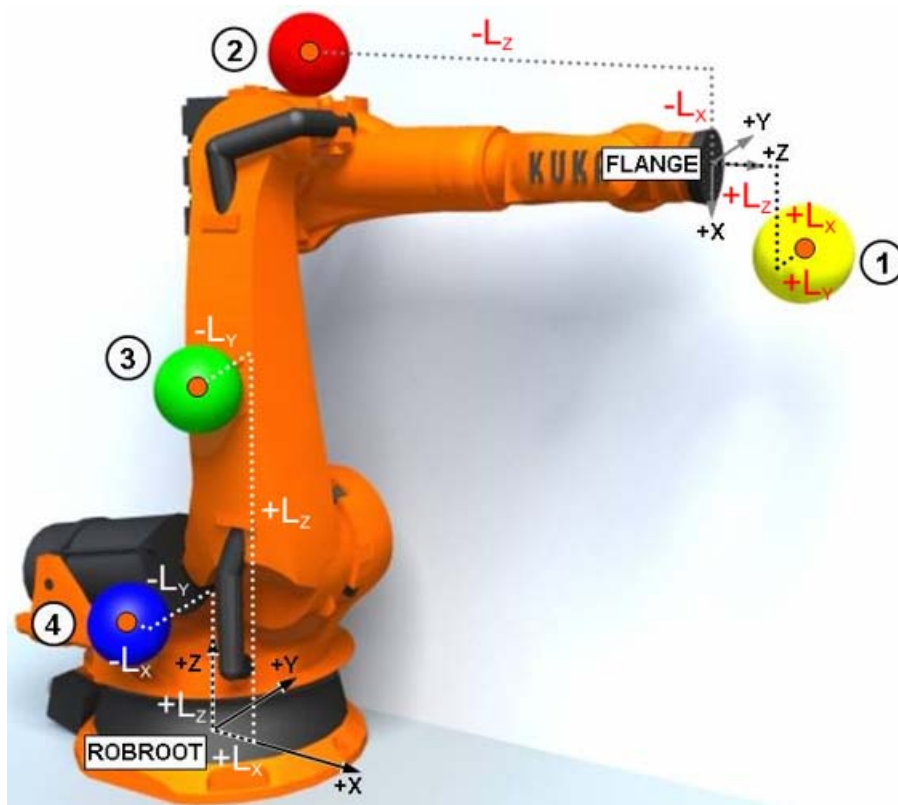


Рис. 7-10: Нагрузки на работе

- |                                    |                                    |
|------------------------------------|------------------------------------|
| 1 Грузоподъемность                 | 3 Дополнительная нагрузка на ось 2 |
| 2 Дополнительная нагрузка на ось 3 | 4 Дополнительная нагрузка на ось 1 |

### 7.5 Данные нагрузки инструмента

Продолжительность: 00:20:00

Оснащение: Модель робота

Информация: Зачем нужны данные нагрузки?

Данные нагрузки учитываются при расчете траекторий и ускорений, а также используются для оптимизации длительности такта. Данные нагрузки должны быть введены в систему управления роботом.

Данные полезной нагрузки необходимо ввести в систему управления роботом и присвоить соответствующему инструменту. Исключение: если данные полезной нагрузки уже были введены программой KUKA.LoadDataDetermination в систему управления роботом, то вводить их вручную больше не нужно.

Сертификат:

## Описание

Каждому геометрически откалиброванному инструменту должны быть присвоены верные данные нагрузки. К этому относятся следующие значения:

- масса ( $m$ ) в кг;
- расстояние от центра тяжести массы до фланца ( $X$ ,  $Y$  и  $Z$ ) в мм;
- ориентация главных осей инерции относительно фланца ( $A$ ,  $B$  и  $C$ ) в градусах ( $^\circ$ );
- моменты инерции массы вокруг осей инерции ( $J_x$ ,  $J_y$  и  $J_z$ ) в  $\text{кгм}^2$ .

Не превышать указанные в технических данных каждого робота максимальные нагрузки!

Что представляют собой моменты инерции?

Момент инерции, также момент инерции масс или инерционный момент, является физической величиной в классической механике. Он указывает сопротивление неподвижного тела изменению своего вращательного движения. При этом он играет роль, аналогичную массе при прямолинейном перемещении. В качестве физической величины он был впервые указан в 1740 г. в работе *Theoria motus corporum solidorum seu rigidorum* Леонарда Эйлера. Момент инерции тела зависит от формы и распределения массы, а также от оси вращения. Поэтому для полного описания действия инерции неподвижного тела недостаточно отдельного числа. Для этого используют датчик инерции. Момент инерции для любой оси можно рассчитать с помощью датчика инерции.

Если привести во вращение тела с различным распределением массы, напр., два шара с равной массой и различным диаметром, один из которых сделан из дерева, а другой – из свинца, то распределение массы вокруг оси вращения будет решающим. Чем дальше части массы будут удалены от оси вращения, тем больше будет (на основе закона рычага) момент вращения, требуемый для приведения обоих шаров в течение определенного времени во вращение с равной частотой. Для тела в качестве суммы своих точек массы следует: для деревянного шара, который имеет большие размеры при равной массе, требуется больший момент инерции. Инерция, которую шары противопоставляют изменению угловой скорости, описывается посредством момента инерции.

С помощью простого эксперимента можно наглядно показать изменение момента инерции. Нужно сесть как можно ближе к середине вращающегося офисного кресла и попытаться повернуться на нем с вытянутыми руками и ногами. Если прижать руки и ноги к телу, то момент инерции уменьшится. Это приведет к тому, что вращательное движение становится быстрее, поскольку импульс вращения остается прежним. Повторное вытягивание рук и ног замедляет перемещение. Чтобы усилить эффект, можно взять в руки тяжелые предметы, например, гантели. Чем больше их масса, тем отчетливее будет эффект. Схожим примером является эффект пируэта, известный из фигурного катания. Контролировать вращательную скорость можно лишь посредством переноса массы тела из оси вращения. Если фигурист вытянет руки или выпрямится из положения группировки, то он будет вращаться быстрее: повторный взмах не требуется.

Самыми употребительными обозначениями момента инерции в формулах являются  $J$  и  $I$ , восходящие к латинскому слову *iners*, которое означает «инертный, медлительный». Единицей измерения момента инерции в системе СИ является  $[\text{кгм}^2]$ .

Зачем нужно задавать значения  $A$ ,  $B$  и  $C$ ?

Значения  $A$ ,  $B$  и  $C$  обозначают положение осей инерции. Термин «главная ось инерции», зачастую сокращенно «главная ось», означают в технической механике особые оси вращения пространственного тела. Обозначение «главная ось инерции» является более частным, поскольку в этом случае имеются в виду три главные оси вращения тела. В теории сопротивления материалов термин «главная ось» обозначает ту ось, для которой плоскостные моменты, или напряжения в математической аналогии, становятся экстремальными. Главные оси всегда находятся вертикально друг к другу и всегда проходят через центр тяжести тела или плоскости. Если при переносе тела его масса является мерой его инерции, то при вращении это место занимает момент инерции. Момент инерции зависит от оси, вокруг которой вращается это тело. В случае с симметричными телами с равномерным распределением массы, например, шаром или кубом, моменты инерции равны для каждой оси, проходящей через центр тяжести, поэтому главные оси не отличаются друг от друга. Оси симметрии всегда являются главными осями. У тел другого вида есть максимальное и минимальное значение. По определению при вращении вокруг оси 1 момент инерции является наибольшим, вокруг оси 3 – наименьшим. При вращении вокруг оси 2 момент инерции не выделяется из общего ряда и находится где-то между наибольшим и наименьшим моментом инерции. Если тело не вращается вокруг одной из этих или параллельной оси, то возникает чисто динамический дисбаланс и может образоваться до трех дополнительных центробежных моментов. Примером этого может служить лопасть винта вертолета, которая вращается вокруг оси винта с небольшим наклоном, а центробежный момент пытается выровнять лопасть винта. Если центробежные моменты равны нулю, то тело вращается вокруг своей главной оси или параллельной ей оси (см. также теорему Штайнера).

**ОСТОРОЖНО**

Эксплуатация робота с неправильными или не действительными данными нагрузки представляет опасность для физического здоровья и жизни персонала и/или может привести к значительному материальному ущербу.

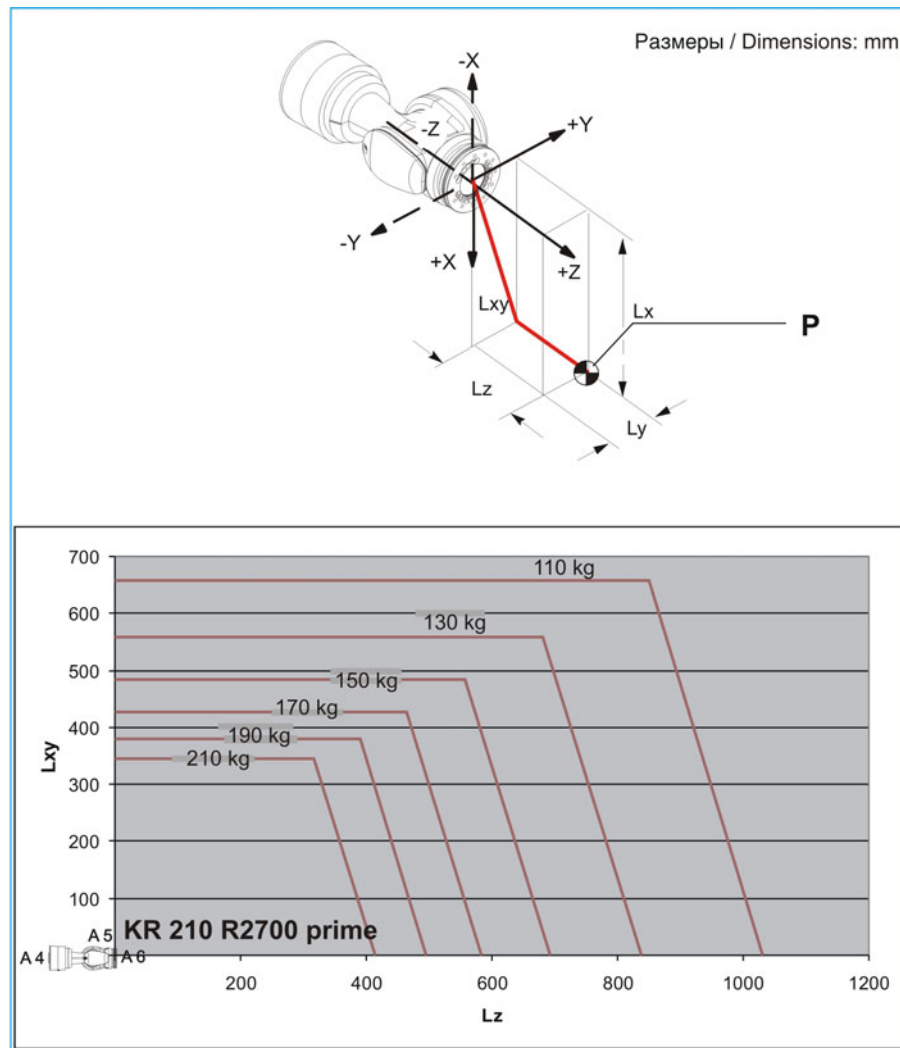


Рис. 7-11: Диаграмма полезной нагрузки

Данные нагрузки можно взять из следующих источников:

- программная опция KUKA.LoadDataDetermination;
- сведения изготовителя;
- расчет вручную;
- программы CAD.

Что представляют собой данные нагрузки инструмента?

Под данными нагрузки инструмента понимают все установленные на фланец робота нагрузки. Они образуют дополнительно установленную на робот массу, которая должна перемещаться вместе с роботом.

К вводимым значениям относятся масса, положение центра тяжести (точка, в которой масса оказывает действие) и моменты инерции масс с относящимися к ним главным осям инерции.

Данные полезной нагрузки **следует** ввести в систему управления роботом и присвоить соответствующему инструменту.

Исключение: если данные полезной нагрузки уже были введены программой KUKA.LoadDataDetermination в систему управления роботом, то вводить их вручную больше не нужно.

**Данные нагрузки инструмента можно взять из следующих источников:**

- опция программного обеспечения KUKA.LoadDetect (только полезные нагрузки);
- сведения изготовителя;
- расчет вручную;
- программы CAD.

**Действие данных нагрузки**

Введенные данные нагрузки оказывают воздействие на большое количество процессов системы управления. К ним относятся:

- алгоритмы системы управления (расчет ускорения);
- контроль скорости и ускорения;
- контроль моментов;
- контроль возможности столкновения;
- контроль энергии;
- и многое другое.

Поэтому крайне важно ввести верные данные нагрузки. Если робот выполняет перемещения с верно установленными данными нагрузки, то:

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

**Порядок действий**

1. В главном меню выбрать пункт **Ввод в эксплуатацию > Калибровка > Инструмент > Данные нагрузки инструмента**.
2. В поле **№ инструмента** ввести номер инструмента. Нажать кнопку **Продолжить** для подтверждения.
3. Ввести данные полезной нагрузки:
  - поле **M**: масса;
  - поля **X, Y, Z**: положение центра тяжести по отношению к фланцу;
  - поля **A, B, C**: ориентация главных осей инерции по отношению к фланцу;
  - поля **JX, JY, JZ**: моменты инерции масс.  
(JX – инерция вокруг оси X системы координат, которая посредством A, B и C повернута по отношению к фланцу. JY и JZ – соответственно, инерции вокруг осей Y и Z).
4. Нажать кнопку **Продолжить** для подтверждения.
5. Нажать кнопку **Сохранить**.

## 7.6 Дополнительные нагрузки на робот

---

Продолжительность: 00:20:00  
 Оснащение: Модель робота  
 Информация:  
 Сертификат:

---

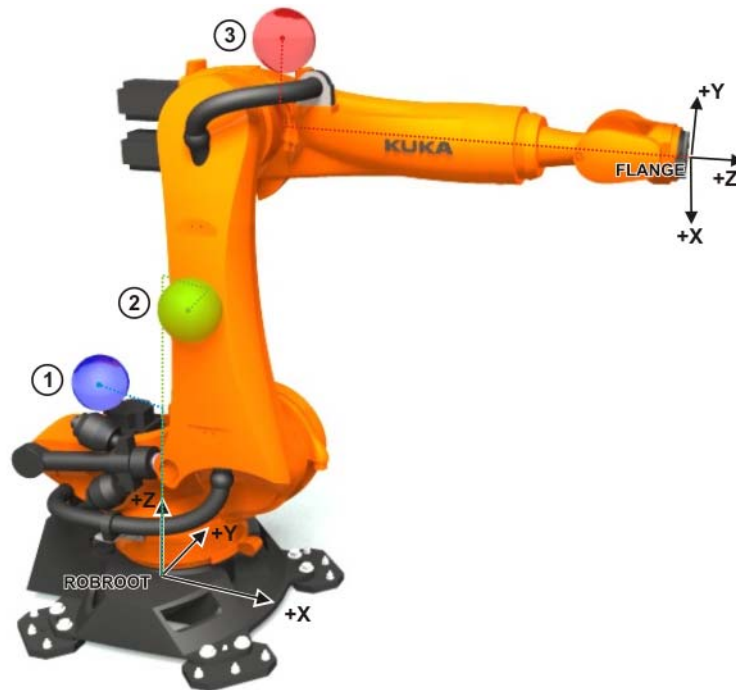
**Дополнительные нагрузки на робот**

Дополнительные нагрузки – дополнительно установленные на станину, балансир или манипулятор компоненты, например:

- система энергоснабжения;
- клапаны;



- система подачи материала;
- запас материала.



**Рис. 7-12: Дополнительные нагрузки на робот**

Данные дополнительной нагрузки должны быть введены в систему управления роботом. К нужным данным относятся:

- масса ( $m$ ) в кг;
- расстояние до центра тяжести массы относительно базовой системы ( $X$ ,  $Y$  и  $Z$ ) в мм;
- ориентация главных осей инерции относительно базовой системы ( $A$ ,  $B$  и  $C$ ) в градусах ( $^\circ$ );
- моменты инерции массы вокруг осей инерции ( $J_x$ ,  $J_y$  и  $J_z$ ) в  $\text{кгм}^2$ .

Базовые системы значений  $X$ ,  $Y$ ,  $Z$  для каждой дополнительной нагрузки:

Нагрузка	Базовая система
Дополнительная нагрузка A1	Система координат ROBROOT $A1 = 0^\circ$
Дополнительная нагрузка A2	Система координат ROBROOT $A2 = -90^\circ$
Дополнительная нагрузка A3	Система координат FLANGE $A4 = 0^\circ$ , $A5 = 0^\circ$ , $A6 = 0^\circ$

Данные нагрузки можно взять из следующих источников:

- опция программного обеспечения KUKA.LoadDetect (только полезные нагрузки);
- сведения изготовителя;
- расчет вручную;
- программы CAD.

**Воздействие дополнительных нагрузок на перемещение робота**

Указание данных нагрузки воздействует на перемещение робота различным образом:

- проектирование траекторий перемещения;
- ускорения;
- время такта;
- износ.



**ОСТОРОЖНО**

Эксплуатация робота с неправильными или не действительными данными нагрузки представляет опасность для физического здоровья и жизни персонала и/или может привести к значительному материальному ущербу.

**Порядок действий**

1. В главном меню выбрать пункт **Ввод в эксплуатацию > Калибровка > Данные дополнительной загрузки**.
2. Ввести номер оси, на которой закреплена дополнительная нагрузка. Нажать кнопку **Продолжить** для подтверждения.
3. Ввести данные нагрузки. Нажать кнопку **Продолжить** для подтверждения.
4. Нажать кнопку **Сохранить**.

## 7.7 Калибровка инструмента

---

Продолжительность: 00:45:00

Оснащение: Модель робота  
Пересечение координат

Информация: Направление удара по оси Z можно активировать в папке C:\KRC\Util\Regutil\....

Сертификат:

---

**Описание**

*Калибровка* инструмента означает, что создается система координат, начало которой находится в точке начала отсчета инструмента. Эту точку начала отсчета называют **TCP** (Tool Center Point), системой координат является система координат **TOOL**.

Калибровка инструмента также включает в себя калибровку:

- точки TCP (начало системы координат);
- ориентации системы координат.



Рис. 7-13: Примеры откалиброванных инструментов



Может быть сохранено до 16 систем координат TOOL. (Переменная: TOOL\_DATA[1...16]).

При калибровке сохраняется расстояние от системы координат TOOL (по оси X, Y и Z) до системы координат фланца, а также их поворот относительно друг друга (угол A, B и C).

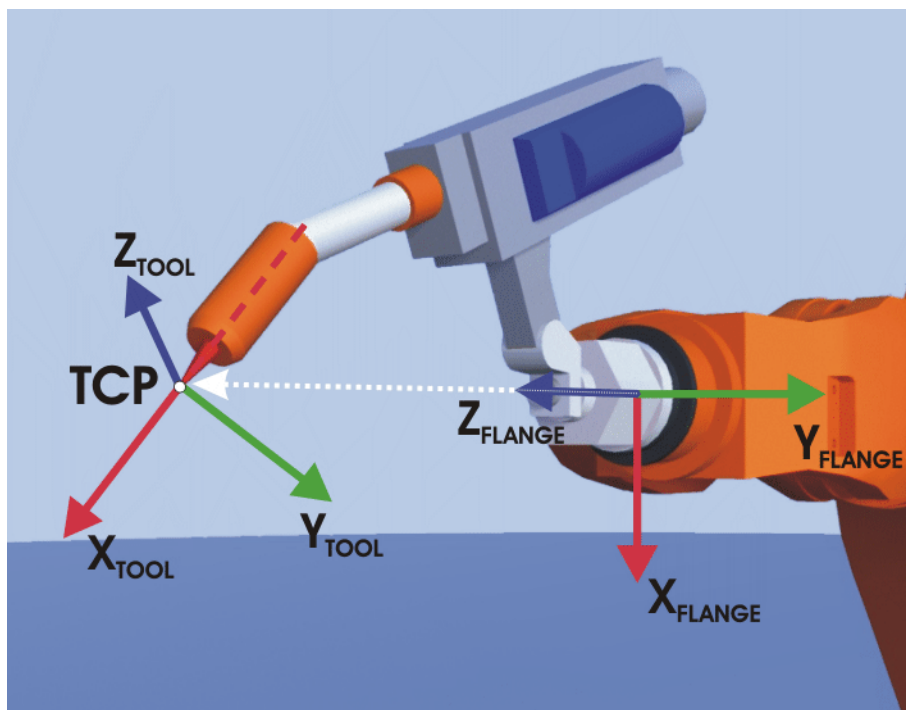


Рис. 7-14: Принцип калибровки TCP

Система координат фланца всегда известна системе управления! Неоткалиброванный инструмент имеет то же положение и ориентацию, что и система координат фланца!

#### Преимущества

Если была выполнена точная калибровка какого-либо инструмента, обслуживающий и программирующий персонал на практике получает следующие преимущества:

- улучшенное перемещение вручную:
  - возможность переориентации TCP (напр., головки инструмента);

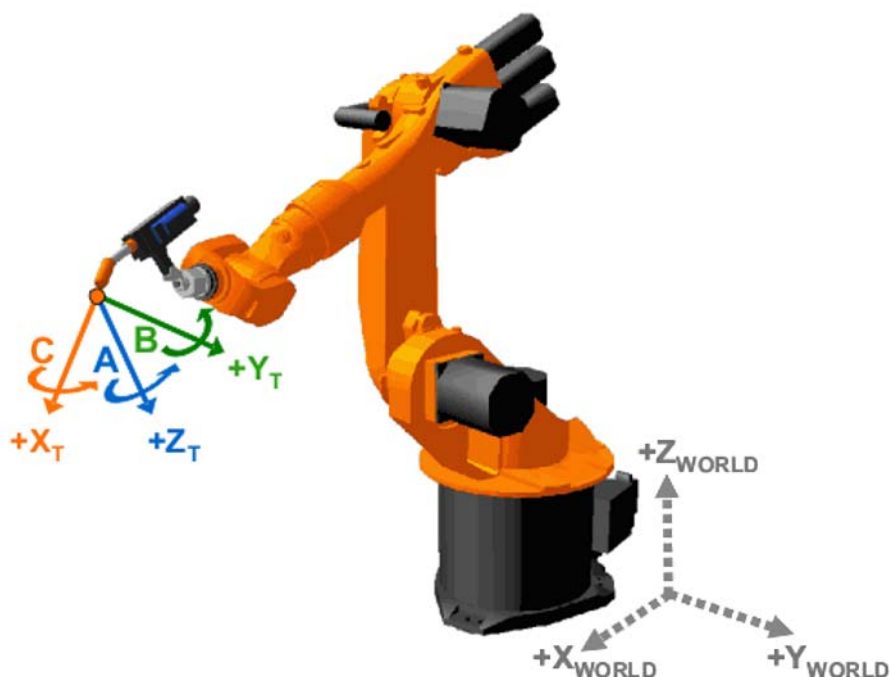


Рис. 7-15: Переориентация вокруг точки TCP

- перемещение в направлении удара инструмента;

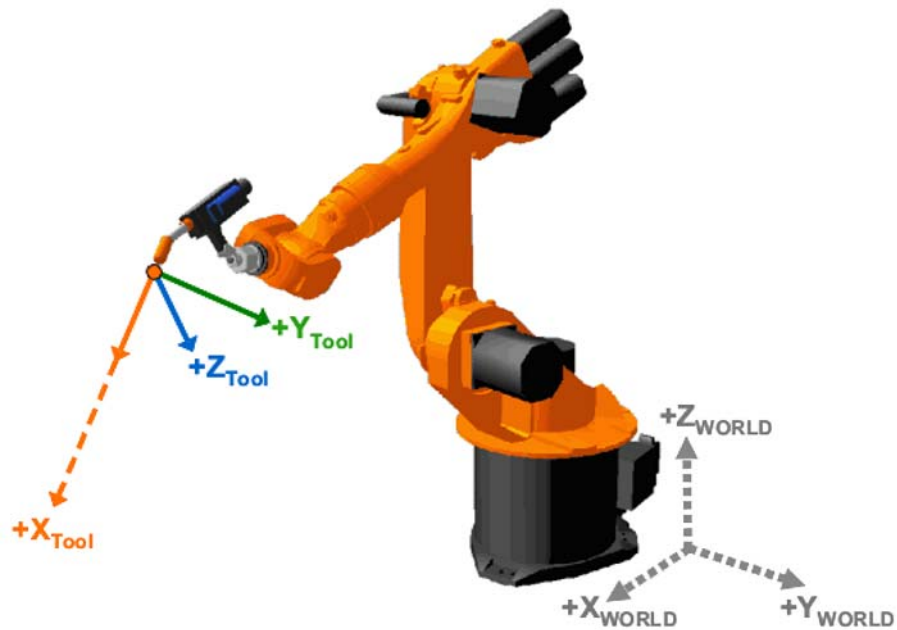


Рис. 7-16: Направление удара TCP

- использование при программировании перемещения;
  - запрограммированная скорость перемещения сохраняется вдоль траектории перемещения у TCP;

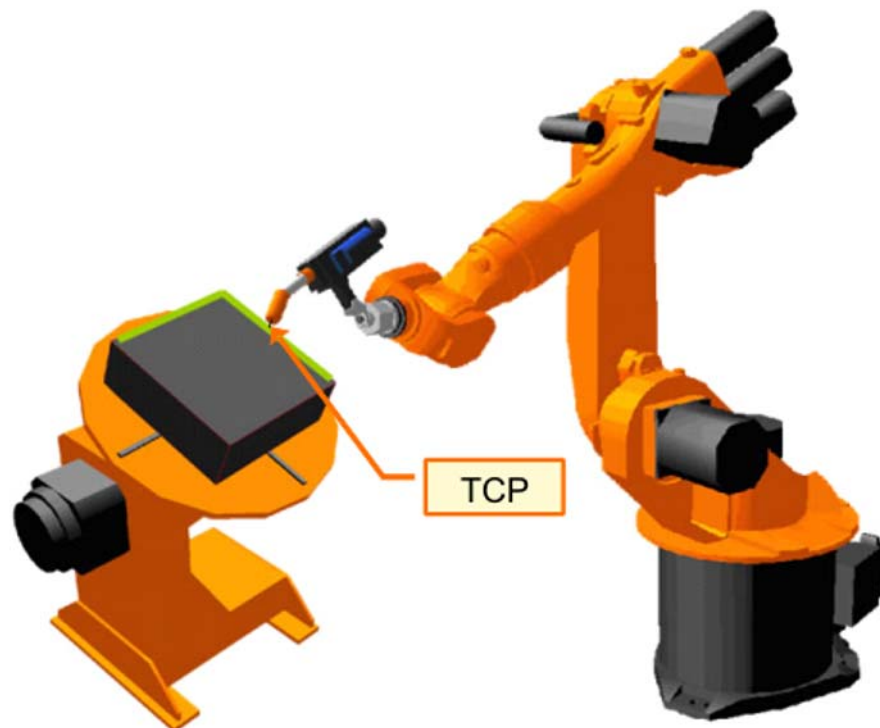


Рис. 7-17: Программный режим с TCP

- кроме того, возможна заданная ориентация вдоль траектории перемещения.

### Возможности калибровки инструмента

Калибровка инструмента состоит из 2 шагов:

Шаг	Описание
1	<p><b>Определение начала системы координат TOOL</b></p> <p>На выбор предлагаются следующие методы:</p> <ul style="list-style-type: none"> <li>■ XYZ, 4 точки;</li> <li>■ база отсчета XYZ.</li> </ul>
2	<p><b>Определение ориентации системы координат TOOL</b></p> <p>На выбор предлагаются следующие методы:</p> <ul style="list-style-type: none"> <li>■ ABC, унив.;</li> <li>■ ABC, 2 точки.</li> </ul>
Альтернатива	<p>Прямой ввод значений для расстояния до центра фланца (X, Y, Z) и поворота (A, B, C):</p> <ul style="list-style-type: none"> <li>■ цифровой ввод.</li> </ul>

### Калибровка TCP с помощью метода «XYZ, 4 точки»

Точку TCP калибруемого инструмента можно подвести к отсчетной точке из 4 различных направлений. Точку начала отсчета можно выбрать произвольно. Система управления роботом рассчитывает точку TCP, исходя из различных положений фланца.



4 положения фланца, из которых выполняется подвод к отсчетной точке, должны находиться на достаточном расстоянии друг от друга.

#### Порядок действий при методе «XYZ, 4 точки»:

1. Выбрать последовательность меню **Ввод в экпл. > Замерить > Инструмент > XYZ, 4 точки**.
2. Присвоить номер и имя калибруемому инструменту. Нажать для подтверждения кнопку **ОК**.



Для выбора доступны номера 1– 16.

3. Подвести точку TCP к отсчетной точке. Нажать для подтверждения кнопку **ОК**.
4. Подвести точку TCP к отсчетной точке с другого направления. Нажать для подтверждения кнопку **ОК**.



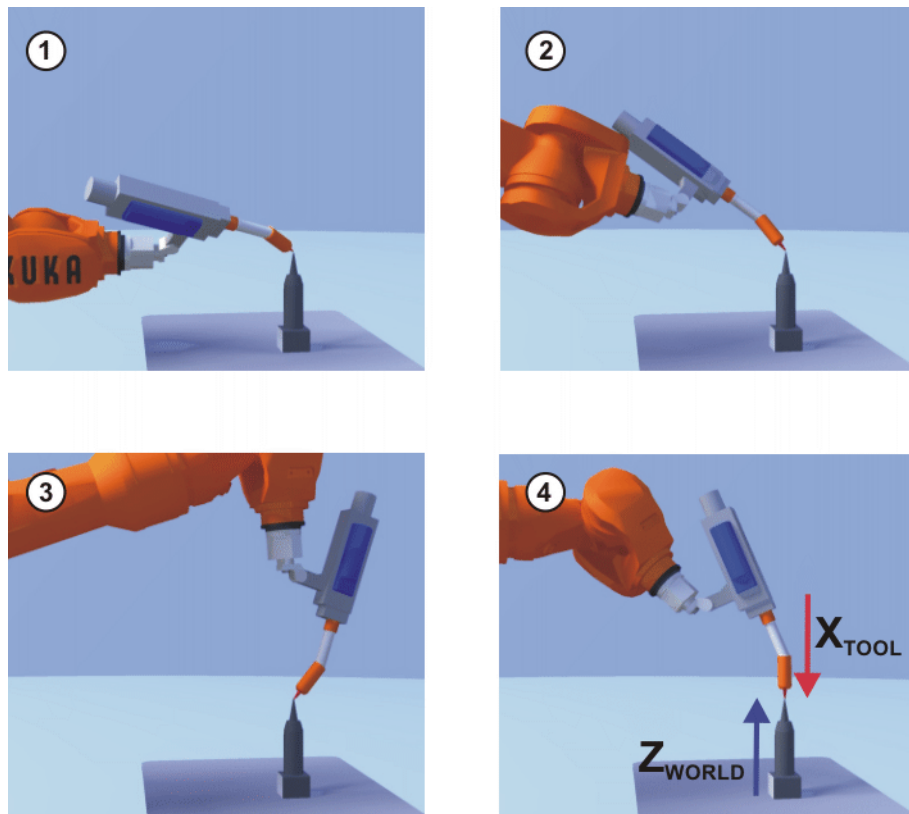


Рис. 7-18: Метод XYZ 4 точки

5. Дважды повторить шаг 4.
6. Нажать кнопку **Сохранить**.

### Калибровка ТСР с помощью базы отсчета XYZ

При использовании базы отсчета XYZ калибровка нового инструмента выполняется с использованием уже откалиброванного инструмента. Система управления роботом сравнивает положения фланца и рассчитывает ТСР нового инструмента.

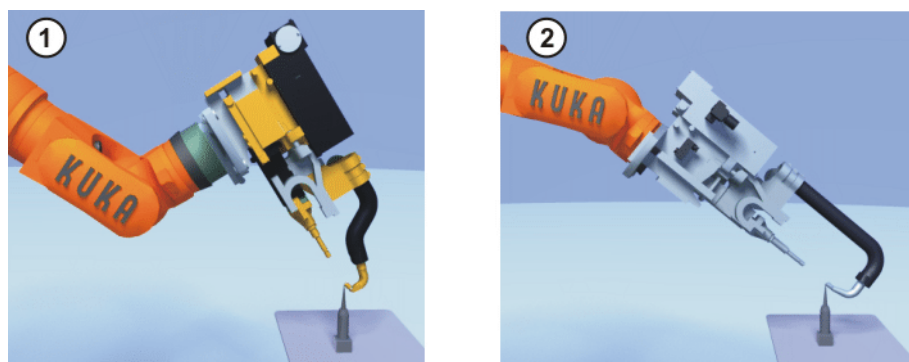


Рис. 7-19

### Порядок действий

1. Условие: на установочном фланце установлен уже откалиброванный инструмент и известны данные точки ТСР.
2. В главном меню выбрать пункт **Ввод в эксплуатацию > Калибровка > Инструмент > База отсчета XYZ**.
3. Присвоить номер и имя новому инструменту. Нажать кнопку **Далее** для подтверждения.
4. Ввести данные ТСР уже откалиброванного инструмента. Нажать кнопку **Далее** для подтверждения.

5. Подвести точку TCP к отсчетной точке. Нажать кнопку **Калибровка**. Нажать кнопку **Далее** для подтверждения.
6. Отвести и снять инструмент. Установить новый инструмент.
7. Посредством TCP нового инструмента выполнить подвод к отсчетной точке. Нажать кнопку **Калибровка**. Нажать кнопку **Далее** для подтверждения.
8. Нажать кнопку **Сохранить**. Данные сохранятся, и окно закроется.

Или же нажать кнопку **Данные нагрузки**. Данные сохранятся, и откроется окно, в котором можно ввести данные полезной нагрузки.

### Метод ориентации и калибровки ABC World

Оси системы координат TOOL устанавливаются параллельно осям системы координат WORLD. Посредством этого системе управления роботом сообщается ориентация системы координат TOOL.

Метод имеет 2 варианта:

- **5D**: системе управления роботом сообщается только направление удара инструмента. Направлением удара является по умолчанию ось X. Направление других осей задается системой, и оператор не может определить их без дополнительных мер.

Область применения: например, дугая сварка в инертном/активном газе, лазерная или гидравлическая резка.

- **6D**: системе управления роботом сообщается направление всех 3 осей.

Область применения: например, для сварочных клещей, захвата или сопел для нанесения клея.

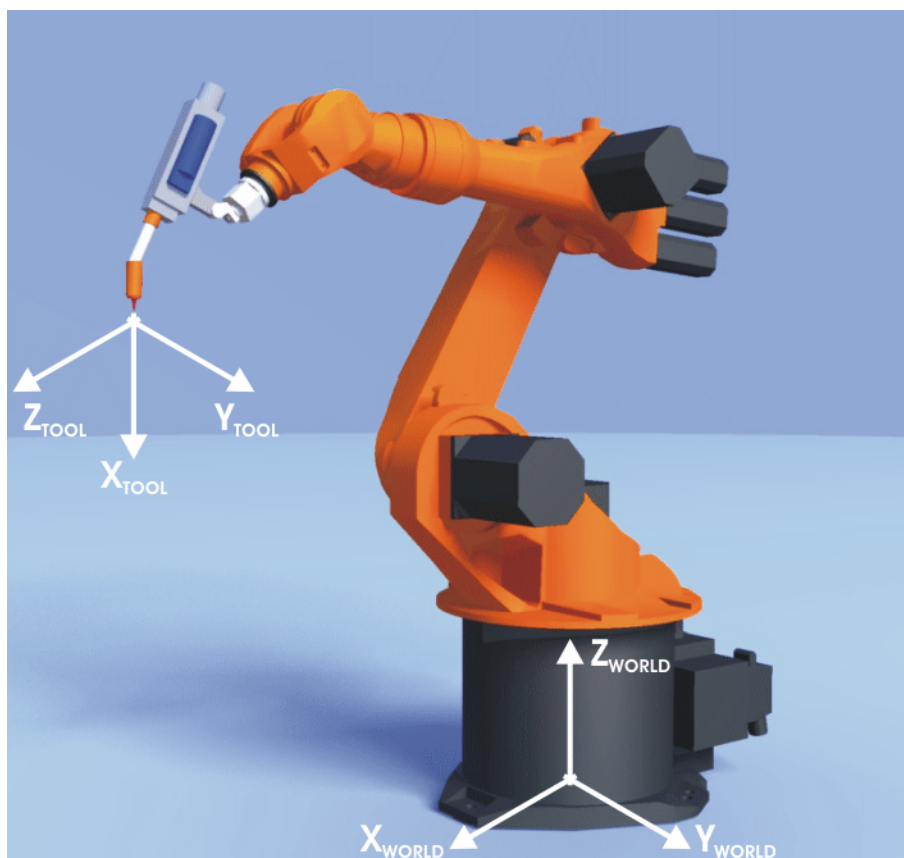


Рис. 7-20: Метод ABC унив.

#### Порядок действий для метода ABC World 5D

- а. Выбрать последовательность меню **Ввод в экспл. > Калибровка > Инструмент > ABC унив..**

- b. Ввести номер инструмента. Нажать для подтверждения кнопку **ОК**.
- c. В поле **5D/6D** выбрать один из вариантов. Нажать для подтверждения кнопку **ОК**.
- d. Если выбрано **5D**:  
 $+X_{TOOL}$  параллельно  $-Z_{WORLD}$  ( $+X_{TOOL}$  = направление удара инструмента).
- e. Нажать для подтверждения кнопку **ОК**.
- f. Нажать кнопку **Сохранить**.

### Калибровка ориентации, метод «ABC,2 точки»

Системе управления роботом сообщаются оси системы координат TOOL путем подвода инструмента к одной точке на оси X и к одной точке в плоскости XY.

Этот метод используется, если направления осей должны быть определены особенно точно.

**i** Соблюдать следующий порядок действий, если направление удара инструмента является направлением по умолчанию (= направление X). Если направление удара изменено на Y или Z, необходимо соответственно изменить порядок действий.

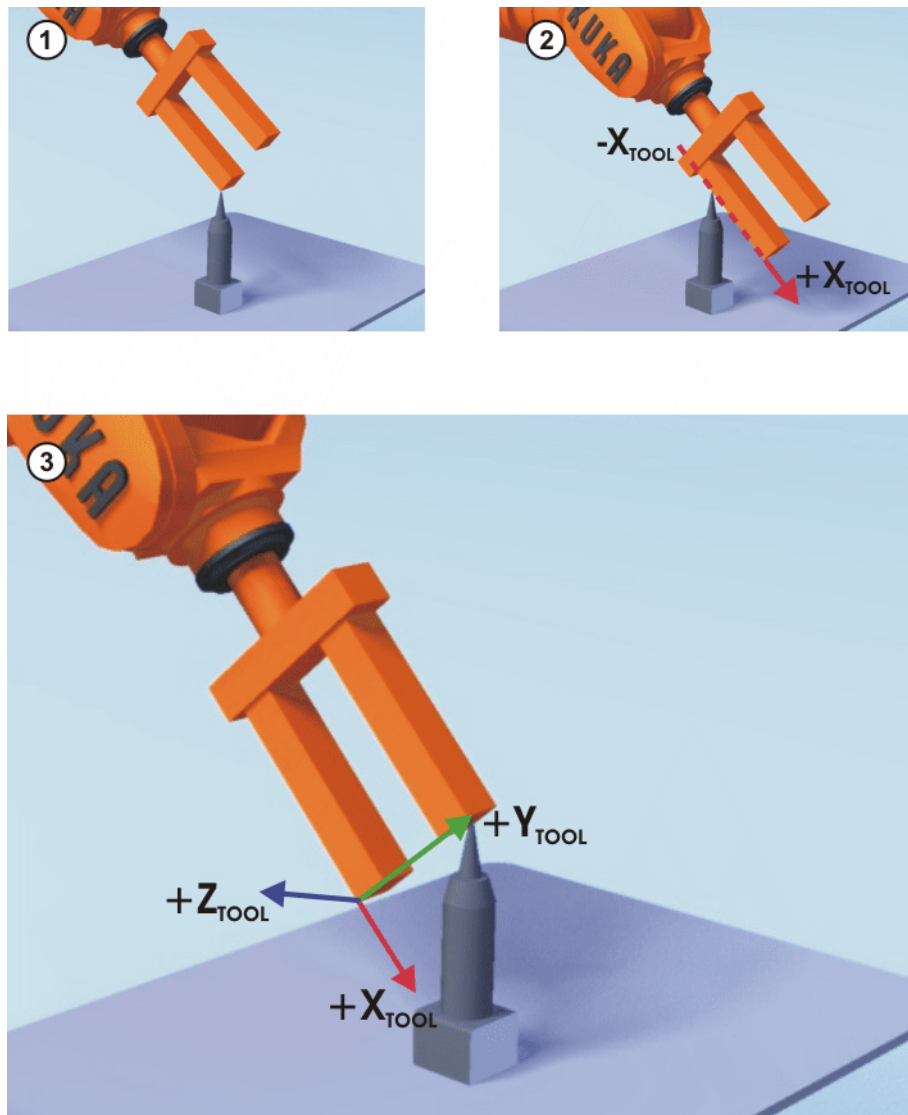


Рис. 7-21: Метод ABC 2 точки

1. Условие заключается в том, чтобы точка TCP уже была откалибрована методом XYZ.
2. В главном меню выбрать пункт **Ввод в эксплуатацию > Калибровка > Инструмент > ABC, 2 точки**.
3. Ввести номер установленного инструмента. Нажать кнопку **Далее** для подтверждения.
4. Посредством TCP подвести робот к любой отсчетной точке. Нажать кнопку **Калибровка**. Нажать кнопку **Далее** для подтверждения.
5. Переместить инструмент таким образом, чтобы отсчетная точка на оси X совпала с точкой с отрицательным значением X (т. е. против направления удара инструмента). Нажать кнопку **Калибровка**. Нажать кнопку **Далее** для подтверждения.
6. Переместить инструмент таким образом, чтобы отсчетная точка на плоскости XY совпала с точкой с положительным значением Y. Нажать кнопку **Калибровка**. Нажать кнопку **Далее** для подтверждения.
7. Нажать кнопку **Сохранить**. Данные сохраняются, и окно закрывается. Или же нажать кнопку **Данные нагрузки**. Данные сохраняются, и открывается окно, в котором можно ввести данные полезной нагрузки.

## 7.8 Калибровка базы

---

Продолжительность: 00:30:00  
 Оснащение:  
 Информация:  
 Сертификат:

---

### Описание

*Калибровка* базы означает создание системы координат в определенном месте окружения робота, исходя из универсальной системы координат. Цель заключается в том, чтобы перемещения и запрограммированные положения робота относились к этой системе координат. Поэтому определенные кромки захватов для заготовок, секций, палет или машин образуют целесообразные отсчетные точки для основной системы координат.

Калибровка базы выполняется в два этапа:

1. определение начала координат;
2. определение направлений координат.

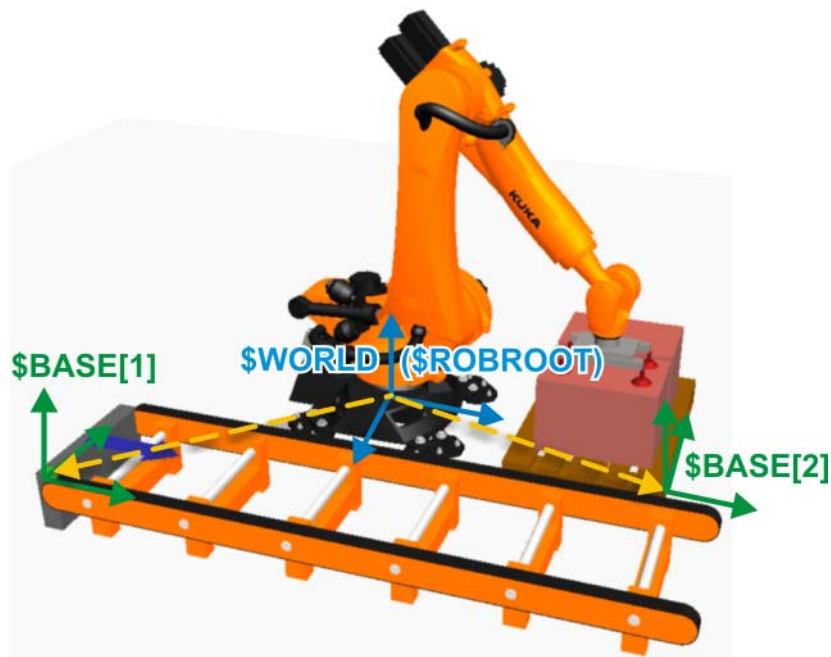


Рис. 7-22: Калибровка базы

**Преимущества**

Выполнение калибровки базы предоставляет следующие преимущества:

- перемещение вдоль кромки заготовки:  
точка TCP может перемещаться вдоль кромок рабочей поверхности или заготовки вручную;

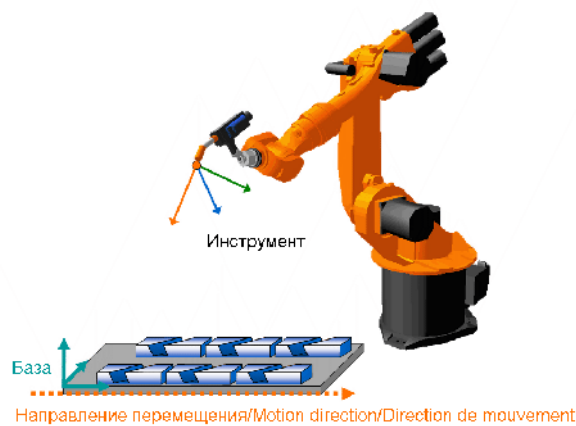
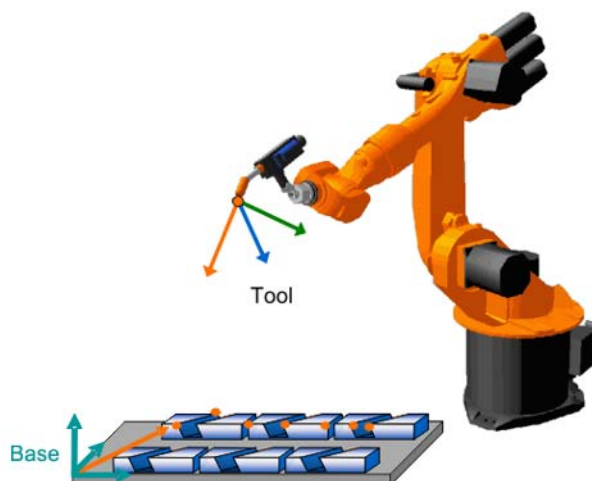


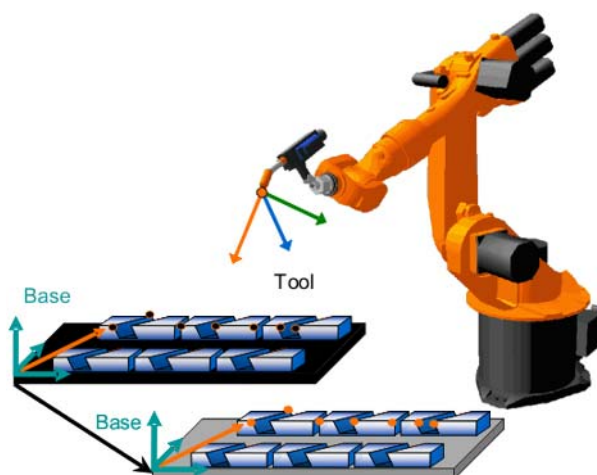
Рис. 7-23: Преимущества калибровки базы: направление перемещения

- система относительных координат:  
заученные точки относятся к выбранной системе координат;



**Рис. 7-24: Преимущества калибровки базы: отношение к нужной системе координат**

- коррекция/смещение системы координат:  
возможно заучивание точек относительно базы. Если необходимо сместить базу, например, из-за перемещения рабочей поверхности, точки смещаются вместе с ней, и новое обучение не требуется;



**Рис. 7-25: Преимущества калибровки базы: смещение основной системы координат**

- использование нескольких основных систем координат:  
можно создать и использовать в зависимости от этапа выполнения программы до 32 разных систем координат.



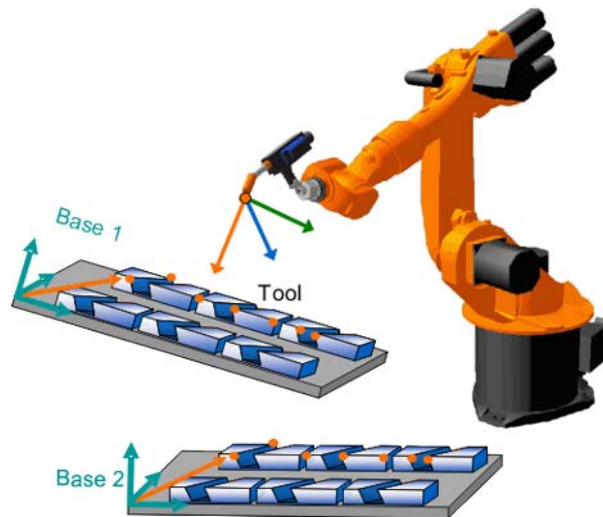


Рис. 7-26: Преимущества калибровки базы: использование нескольких основных систем координат

### Возможности калибровки базы

Для выбора доступны следующие методы калибровки базы:

Метод	Описание
<b>Метод 3 точек</b>	<ol style="list-style-type: none"> <li>1. Определение начала координат.</li> <li>2. Определение положительного направления X.</li> <li>3. Определение положительного направления Y (плоскость XY).</li> </ol>
<b>Косвенный метод</b>	<p>Косвенный метод используется в случае, если подвод к начальной точке базы невозможен, например, так как она расположена внутри заготовки или вне рабочего пространства робота.</p> <p>Выполняется подвод к 4 точкам базы, координаты которых должны быть известны (данные CAD). Система управления роботом рассчитывает базу на основании этих точек.</p>
<b>Цифровой ввод</b>	<p>Прямой ввод значений для расстояния относительно универсальной системы координат (X, Y, Z) и поворота (A, B, C).</p>

### Порядок действий для метода 3 точек



Калибровку базы можно выполнить лишь с заранее откалиброванным инструментом (точка TCP должна быть известна).

1. В главном меню выбрать пункт **Пуск в эксплуатацию > Калибровка > База > 3 точки**.
2. Присвоить номер и имя базе. Для подтверждения нажать кнопку **Далее**.
3. Ввести номер инструмента, точка TCP которого будет использоваться для калибровки базы. Для подтверждения нажать кнопку **Далее**.
4. Посредством TCP выполнить подвод к началу системы координат новой базы. Нажать программируемую клавишу **Калибровка** и кнопку **Да** для подтверждения положения.



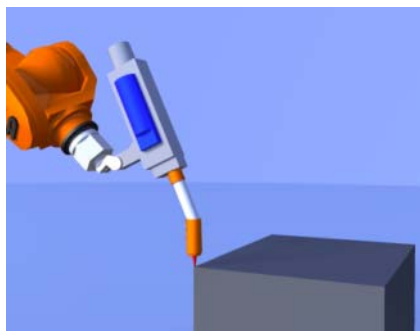


Рис. 7-27: Первый пункт: начало системы координат.

5. С помощью ТСП выполнить подвод к точке на положительной оси X новой базы. Нажать кнопку **Калибровка** и кнопку **Да** для подтверждения положения.

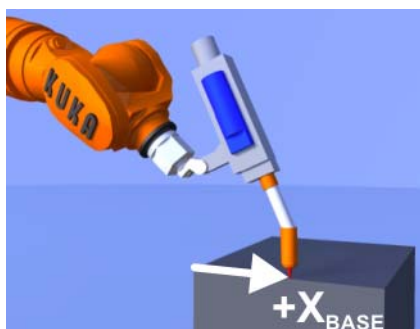


Рис. 7-28: Вторая точка: направление X

6. С помощью ТСП на плоскости XY выполнить подвод к точке с положительным значением Y. Нажать кнопку **Калибровка** и кнопку **Да** для подтверждения положения.

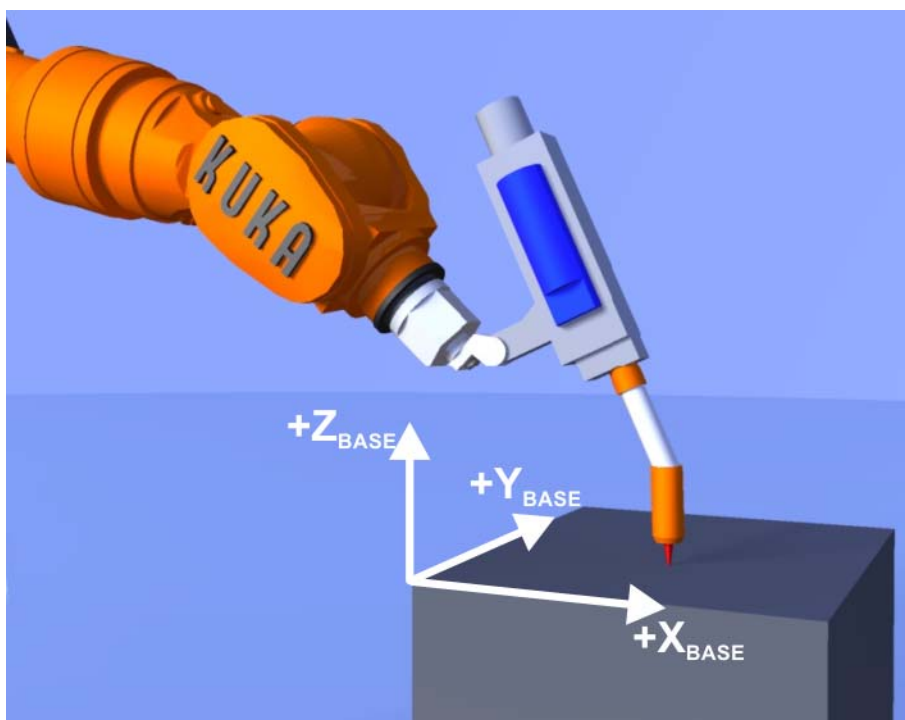


Рис. 7-29: Третья точка: плоскость XY

7. Нажать кнопку **Сохранить**.
8. Закрыть меню.



Три точки калибровки не должны лежать на одной прямой. Между точками должен лежать минимальный угол (стандартная настройка – 2,5°).

## 7.9 Запрос текущего положения робота

Продолжительность: 00:10:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

### Возможности индикации положений робота

Текущее положение робота можно представить двумя различными способами:

- По осям:

$\$AXIS\_ACT=\{A1...,A2...,A3...,A4...,A5...,A6...,E1...,...,E6...\}$

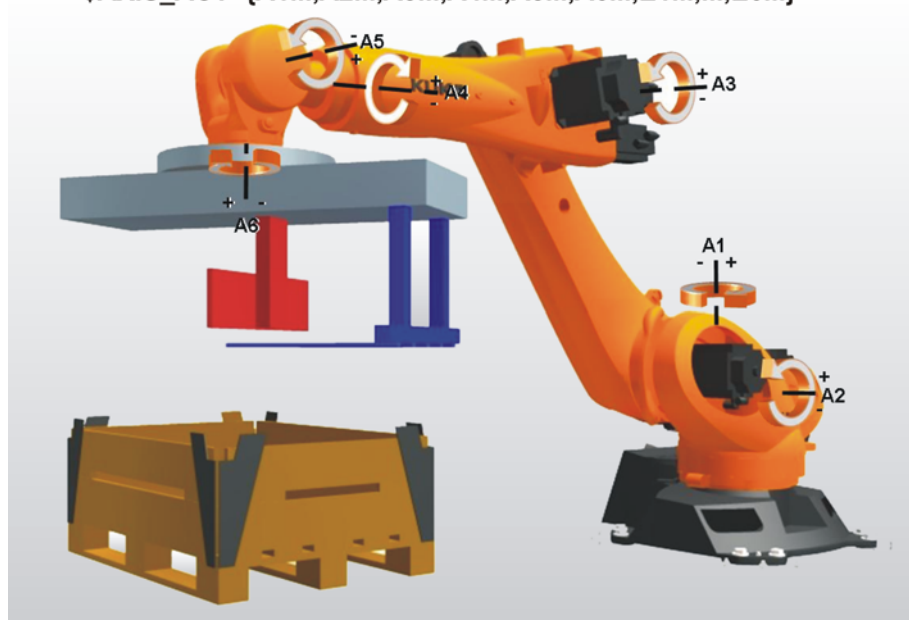


Рис. 7-30: Осевое положение робота

Для каждой оси показывается текущий угол оси: он соответствует абсолютному значению угла, исходя из положения юстировки.

- Декартовый:

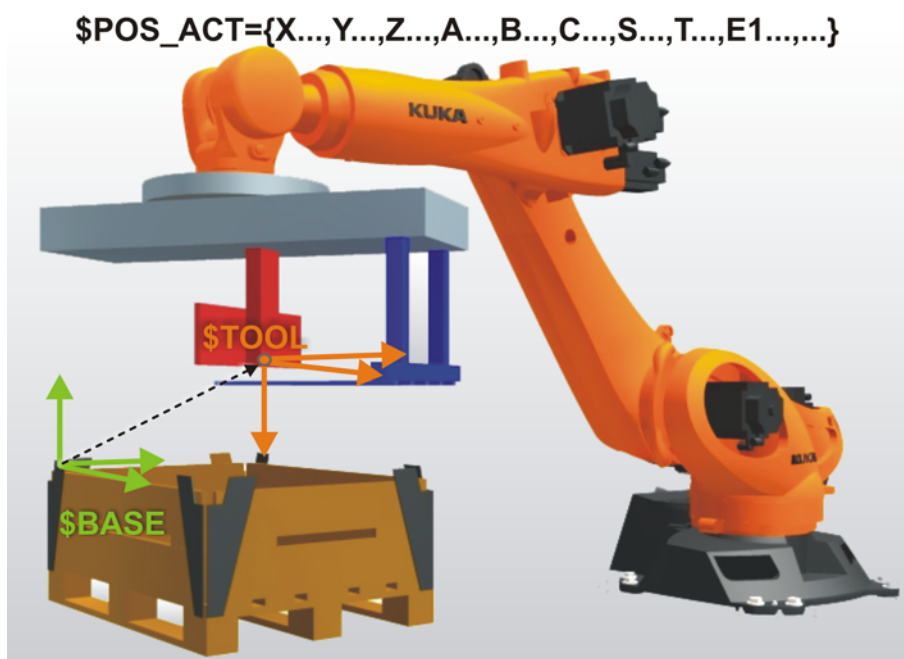


Рис. 7-31: Положение в прямоугольных координатах

Показывается текущее положение текущей точки TCP (системы координат инструмента) по отношению к выбранной в настоящий момент основной системе координат.

Если не выбрана никакая система координат инструмента, действует система координат фланца.

Если не выбрана никакая основная система координат, действует универсальная система координат.

**Положение в прямоугольных координатах с различными основными системами координат**

При рассмотрении приведенного ниже рисунка сразу заметно, что робот трижды принимает одинаковое положение. Однако индикация положения в каждом из трех случаев указывает различные значения:

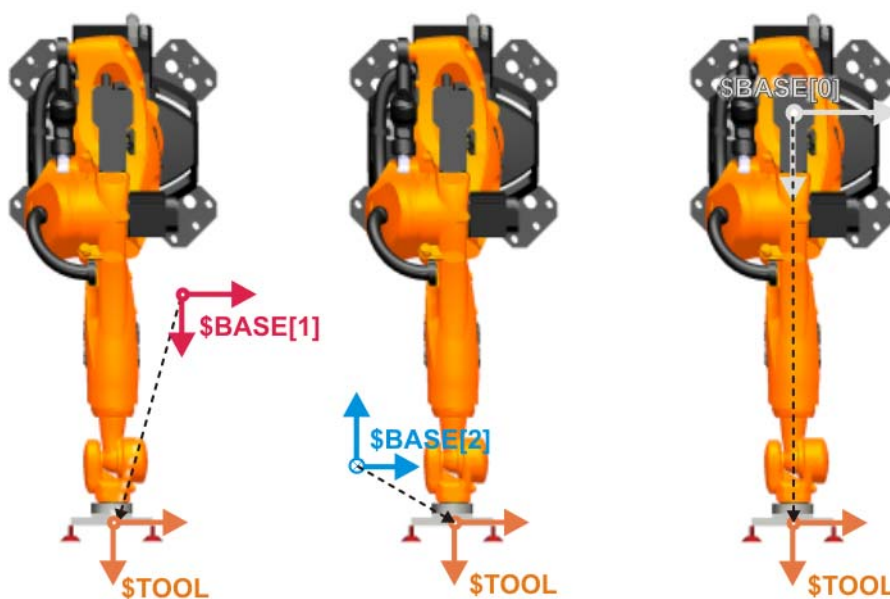


Рис. 7-32: Три положения робота — одно положение робота!

Показывается положение системы координат Tool/точки TCP в текущей основной системе координат:

- для базы 1;

- для базы **2**;
- для базы **0**: это соответствует системе координат основания робота (в большинстве случаев также универсальной системе координат).



**Рис. 7-33: Выбор инструмента и базы**

После выбора верной базы и верного инструмента индикация фактического положения в прямоугольных координатах укажет разумные значения.

### Запрос положения робота

#### Порядок действий:

- В меню выбрать пункт **Индикация > Фактическое положение**. На экране появится фактическое положение в прямоугольной системе координат.
- Чтобы вывести на экран фактическое положение согласно осям, нажать кнопку **Соотв. осям**.
- Чтобы вывести на экран фактическое положение в прямоугольной системе координат, нажать кнопку **В прямоугольных координатах**.



## 8 Выполнение программы робота

### 8.1 Обзор

В этом учебном разделе рассматриваются следующие элементы:

- выполнение перемещения инициализации;
- выбор и запуск программ робота.



Рис. 8-1: Смена главы

### 8.2 Выполнение перемещения инициализации

---

Продолжи-  
тельность: 00:20:00  
Оснащение:  
Информация:  
Сертификат:

---

#### Перемещение SAK

Перемещение инициализации робота KUKA называется перемещением SAK.

**i** SAK означает точное совпадение записи. Совпадение означает «соответствие», а также «совпадение временных/пространственных событий».

Перемещение SAK выполняется в следующих случаях:

- выбор программы;
- сброс программы;
- перемещение вручную во время работы программного режима;
- изменение в программе;
- выбор записи.

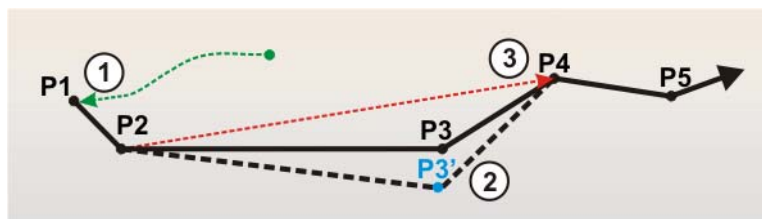


Рис. 8-2: Примеры причин для перемещения SAK

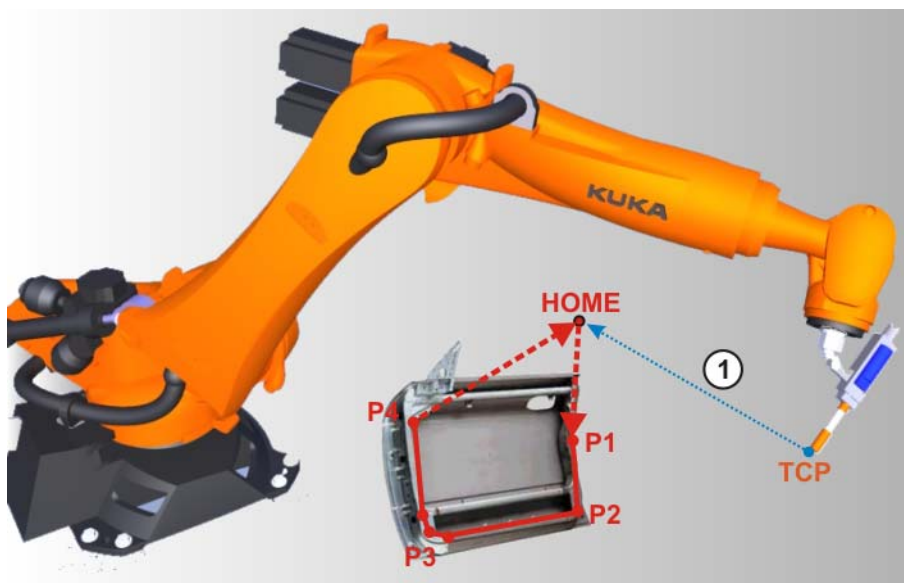
Примеры выполнения перемещения SAK:

- 1 Перемещение SAK в положение Home после выбора или сброса программы.
- 2 Перемещение SAK после изменения команды перемещения: удаление, обучение точки и т. д.
- 3 Перемещение SAK после выбора записи.

**Причины для перемещения SAK**

Перемещение SAK требуется для обеспечения согласования текущего положения робота с координатами текущей точки в программе робота.

Планирование траектории возможно только после того, как текущее положение робота будет равно запрограммированному положению. Сначала всегда следует привести точку TCP на траекторию.



**Рис. 8-3: Пример перемещения SAK**

- 1 Перемещение SAK в положение Home после выбора или сброса программы.

**8.3 Выбор и запуск программ робота**

---

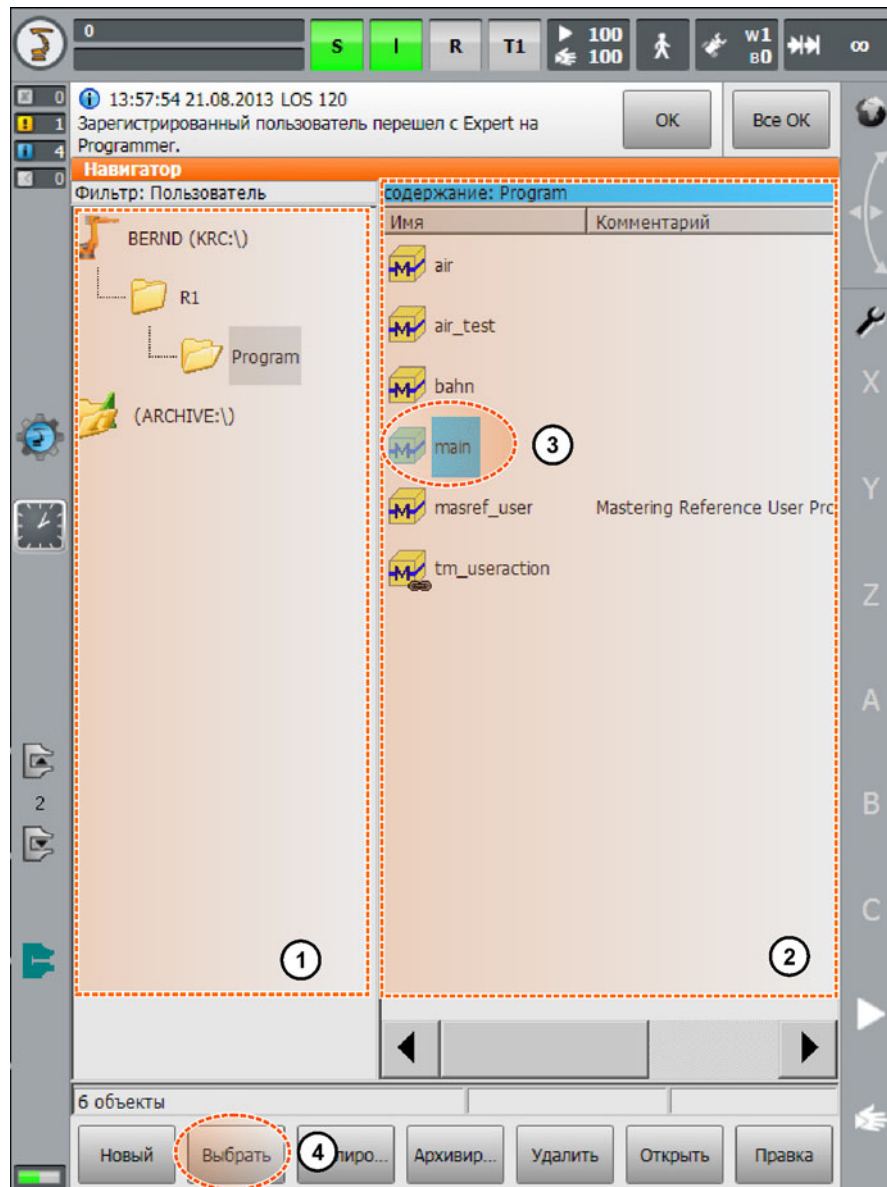
Продолжительность: 00:30:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

---

**Выбор и запуск программ робота**

Чтобы обработать программу робота, ее следует выбрать. Программы робота доступны в пользовательском интерфейсе в навигаторе. Обычно программы перемещения хранятся в папках. Программа Cell (управляющая программа для управления роботом со стороны ПЛК) всегда находится в папке R1.





**Рис. 8-4: Навигатор**




- 1 Навигатор: структура каталогов/накопителей
- 2 Навигатор: списки каталогов/файлов
- 3 Выбранная программа
- 4 Кнопка для выбора программы

Для запуска программы доступны как кнопки запуска в прямом направлении (▶), так и запуска в обратном направлении (◀).



**Рис. 8-5: Направления выполнения программы: вперед/назад**

При обработке программы для управляемого программой перемещения робота доступно несколько **режимов выполнения программы**:

	<p><b>GO</b></p> <ul style="list-style-type: none"> <li>■ Программа последовательно выполняется до ее конца.</li> <li>■ В тестовом режиме кнопка запуска должна удерживаться нажатой.</li> </ul>
	<p><b>Перемещение</b></p> <ul style="list-style-type: none"> <li>■ В режиме выполнения Motion Step каждая команда перемещения обрабатывается отдельно.</li> <li>■ После завершения перемещения следует повторно нажать кнопку запуска.</li> </ul>
	<p><b>Отдельный шаг</b>   Доступен только в группе пользователей «Эксперты»!</p> <ul style="list-style-type: none"> <li>■ В режиме Incremental Step обрабатывается строка за строкой (вне зависимости от содержания строки).</li> <li>■ После каждой строки следует повторно нажимать кнопку запуска.</li> </ul>

Как выглядит программа робота?

```

1 DEF kuka_rocks( )
2 INI
3 PTP HOME Vel= 100 % DEFAULT
4 PTP P1 Vel=100 % PDAT1 Tool[1] Base[0]
5 PTP P2 Vel=100 % PDAT2 Tool[1] Base[0]
6 PTP P3 Vel=100 % PDAT3 Tool[1] Base[0]
7 OUT 1'' State=TRUE CONT
8 LIN P4 Vel=2 m/s CPDAT1 Tool[1] Base[0]
9 PTP HOME Vel= 100 % DEFAULT
10 END



```

Рис. 8-6: Структура программы робота

Видно только для группы пользователей «Эксперты»:	
1	<ul style="list-style-type: none"> <li>«DEF <i>название_программы()</i>» всегда находится в начале программы;</li> <li>«END» указывает на конец программы.</li> </ul>
2	<ul style="list-style-type: none"> <li>строка «INI» содержит вызовы стандартных параметров, которые требуются для верного выполнения программы;</li> <li>строку «INI» всегда следует выполнять первой.</li> </ul>
3	<ul style="list-style-type: none"> <li>собственный текст программы с командами перемещения, командами ожидания/логическими командами и т. д.;</li> <li>команда перемещения «PTP Home» часто применяется в начале или в конце программы, поскольку она является однозначным и известным положением.</li> </ul>

Состояние программы

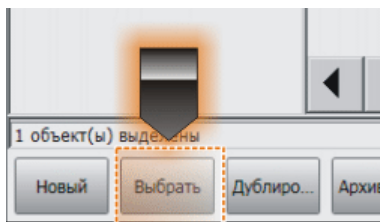
Символ	Цвет	Описание
	Серый	Ни одна из программ не выбрана.
	Желтый	Указатель записи стоит в первой строке выбранной программы.
	Зеленый	Программа выбрана и обрабатывается.

Символ	Цвет	Описание
	Красный	Выбранная и запущенная программа остановлена.
	Черный	Указатель записи находится в конце выбранной программы.

**Запуск программы**

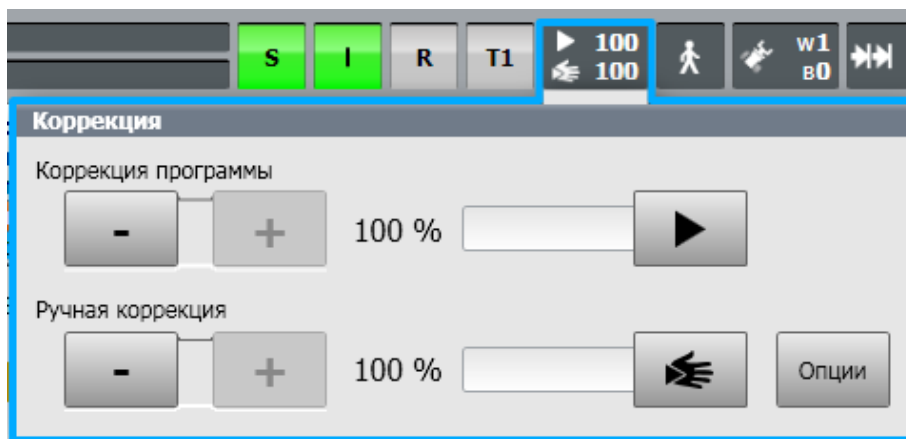
**Порядок действий для запуска программ робота:**

1. Выбрать программу.



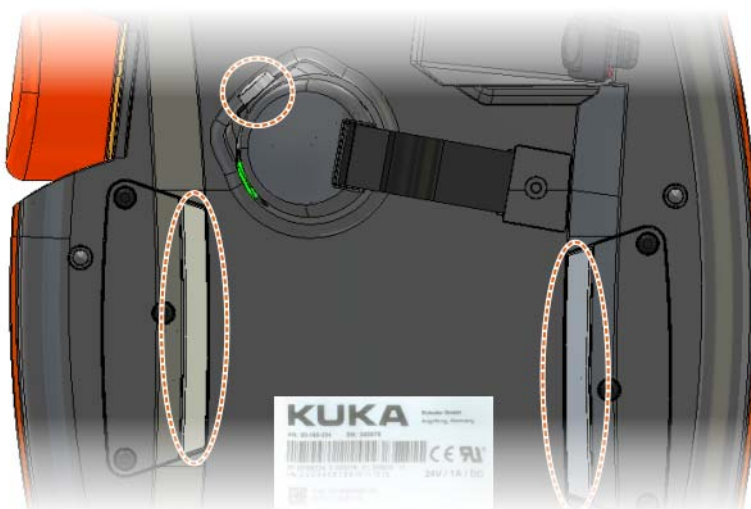
**Рис. 8-7: Выбор программы**

2. Установить скорость программы (коррекция программы, POV).



**Рис. 8-8: Настройка POV**

3. Задействовать клавишу подтверждения.



**Рис. 8-9: Кнопка подтверждения**

4. Нажать и удерживать кнопку запуска (+):

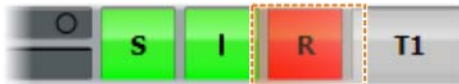
- обрабатывается строка INI;
- робот выполняет перемещение SAK.



Рис. 8-10: Направления выполнения программы: вперед/назад

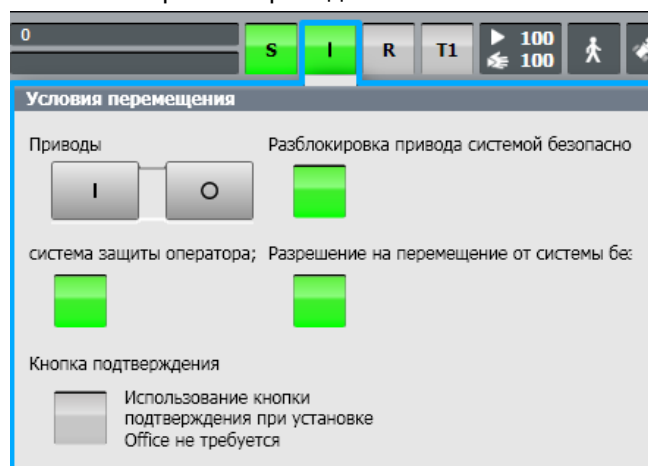
**⚠ ОСТОРОЖНО** Перемещение SAK выполняется в качестве перемещения PTP от фактического положения к целевому положению, если выбранная запись перемещений содержит команду перемещения PTP. Если выбранная запись перемещений содержит LIN или CIRC, то перемещение SAK выполняется в качестве перемещения LIN. Необходимо следить за перемещением, чтобы избежать столкновений. При перемещении SAK скорость автоматически снижается.

5. После достижения целевого положения перемещение останавливается.



Показывается информационное сообщение *SAK достигнут*.

6. Дальнейший порядок выполнения (в зависимости от установленного режима работы):
  - **T1 и T2**: продолжить выполнение программы нажатием на кнопку запуска;
  - **AUT**: активировать приводы.



Затем запустить программу с импульсом нажатием на кнопку *Запуск*.

- В программе Cell переключить режим работы на **EXT** и передать команду перемещения от ПЛК.

## 9 Обращение с файлами программы

### 9.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Создание программных модулей
- Обработка программ робота



Рис. 9-1: Смена главы

### 9.2 Создание программных модулей

Продолжи-  
тельность: 00:20:00  
Оснащение: Офисный ПК  
Информация:  
Сертификат:

#### Программные модули в навигаторе

Программные модули всегда следует помещать в папку «Program». Также можно создать новую папку и сохранять программные модули в ней. Модули обозначены символом с буквой «М». Модуль можно дополнить комментарием. Такой комментарий может содержать, например, краткое описание функции программы.

[Показать на офисном ПК](#)

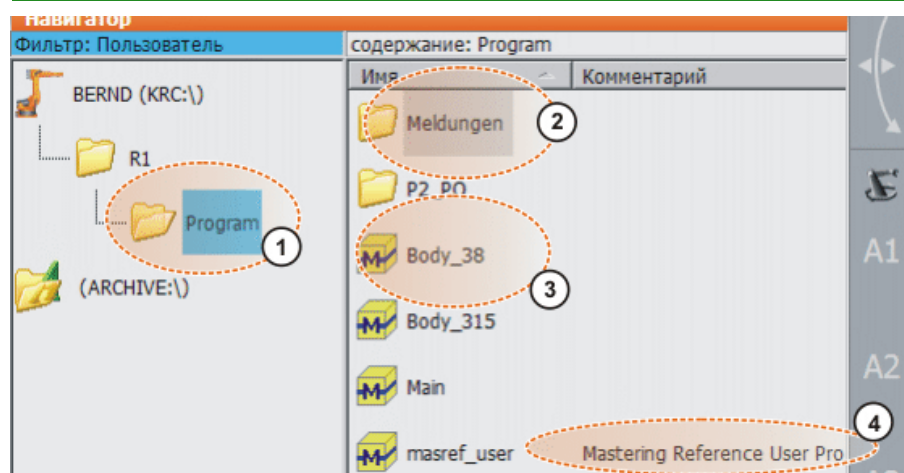


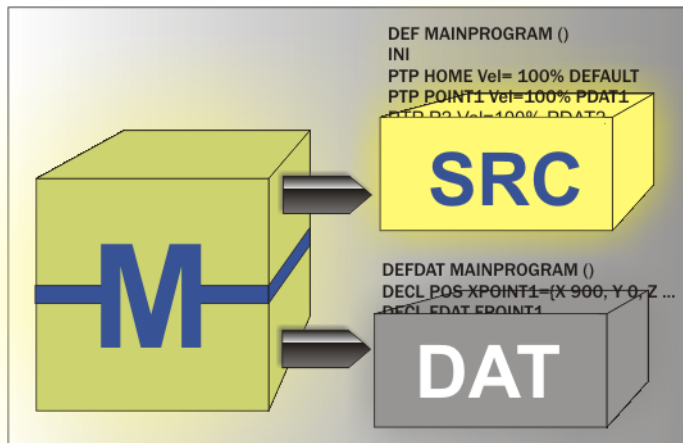
Рис. 9-2: Модули в навигаторе



- 1 Главная папка для программ: «Program».
- 2 Подпапка для других программ.
- 3 Программный модуль/модуль.
- 4 Комментарий к программному модулю.

**Свойства программных модулей**

Модуль всегда состоит из двух частей:



**Рис. 9-3: Структура модуля**

- **исходный код:** файл SRC содержит программный код.

```

DEF MAINPROGRAM ()
INI
PTP HOME Vel= 100% DEFAULT
PTP POINT1 Vel=100% PDAT1 TOOL[1] BASE[2]
PTP P2 Vel=100% PDAT2 TOOL[1] BASE[2]
...
END

```

- **список данных:** файл DAT содержит постоянные данные и координаты точки.

```

DEFDAT MAINPROGRAM ()
DECL_E6POS XPOINT1={X 900, Y 0, Z 800, A 0, B 0, C 0, S 6, T 27, E1
0, E2 0, E3 0, E4 0, E5 0, E6 0}
DECL_FDAT FPOINT1 ...
...
ENDDAT

```

Показать участникам, как можно попасть в файл SRC и список данных (офисный ПК).

**Порядок действий для создания программных модулей**

1. Выделить папку в структуре папок, в которой следует создать новую программу, например, папку **Program**.
2. Нажать программируемую клавишу **Новая**.
3. Ввести имя новой программы и комментарий при необходимости, после чего нажать кнопку **ОК** для подтверждения.

### 9.3 Обработка программных модулей

---

Продолжительность: 00:20:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

---

#### Возможности обработки


Как и в случае с распространенными файловыми системами, программные модули также можно обрабатывать в навигаторе KUKA smartPad.

К обработке относятся:

- дублирование/копирование;
- удаление;
- переименование.


#### Порядок действий для удаления программы

1. В структуре каталогов выделить папку, в которой находится нужный файл.
2. Выделить файл в списке.
3. Нажать программируемую клавишу **Стереть**.
4. Подтвердить операцию, ответив **Да** на контрольный запрос. Модуль будет удален.

 В группе пользователей «Эксперты» и настройке фильтра «Подобно» в навигаторе для каждого модуля показано по два файла (файл SRC и DAT). В таком случае следует удалить оба файла. Удаленные файлы восстановить невозможно.


#### Порядок действий для переименования программы

1. В структуре каталогов выделить папку, в которой находится нужный файл.
2. Выделить файл в списке.
3. Выбрать программируемую клавишу **Обработать > Переименовать**.
4. Поверх имени файла написать новое имя файла и нажать кнопку **OK** для подтверждения.

 В группе пользователей «Эксперты» и настройке фильтра «Подобно» в навигаторе для каждого модуля показано по два файла (файл SRC и DAT). В таком случае следует переименовать оба файла.

#### Порядок действий для дублирования программы

1. В структуре каталогов выделить папку, в которой находится нужный файл.
2. Выделить файл в списке.
3. Нажать программируемую клавишу **Дублировать**.
4. Ввести новое имя файла для модуля и подтвердить его нажатием на кнопку **OK**.

 В группе пользователей «Эксперты» и настройке фильтра «Подобно» в навигаторе для каждого модуля показано по два файла (файл SRC и DAT). В таком случае следует дублировать оба файла.



## 10 Создание и изменение запрограммированных перемещений

### 10.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Создание перемещения с оптимизацией продолжительности такта
- Создание перемещения по траектории
- Изменение команд перемещения



Рис. 10-1: Смена главы

### 10.2 Создание новой команды перемещения

---

Продолжительность:	00:20:00
Оснащение:	Офисный ПК
Информация:	
Сертификат:	

---

Программирование перемещений робота

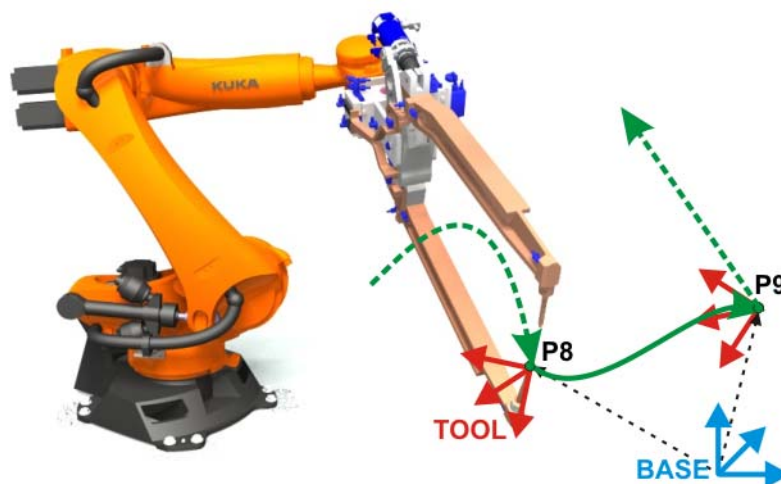


Рис. 10-2: Перемещение робота

Если требуется запрограммировать перемещение робота, возникает много вопросов:

Вопрос	Решение	Ключевое слово
Как робот отмечает для себя свои положения?	Текущее положение инструмента в пространстве сохраняется в памяти (положение робота согласно установленному инструменту и базе).	POS E6POS
Откуда робот знает, как он должен перемещаться?	Посредством указания вида перемещения: от точки к точке, линейно или по кругу.	PTP LIN CIRC
Насколько быстро перемещается робот?	Скорость между двумя точками и ускорение указываются при программировании.	Vel. Acc.
Должен ли робот останавливаться в каждой точке?	Чтобы сэкономить тактовое время, можно также выполнить сглаживание точек; в этом случае точный останов не выполняется.	CONT
Какую ориентацию примет инструмент при достижении какой-либо точки?	Для каждого перемещения можно отдельно установить контроль ориентации.	ORI_TYPE
Определяет ли робот препятствия?	Нет, робот точно следует по запрограммированной траектории. Ответственность за отсутствие столкновений лежит на программисте.  Однако поддерживается контроль возможности столкновения для защиты машины.	Контроль возможности столкновения

Эту информацию следует передать при программировании перемещений робота в режиме обучения. Для этого применяются встроенные формуляры, в которые можно удобно ввести эту информацию.

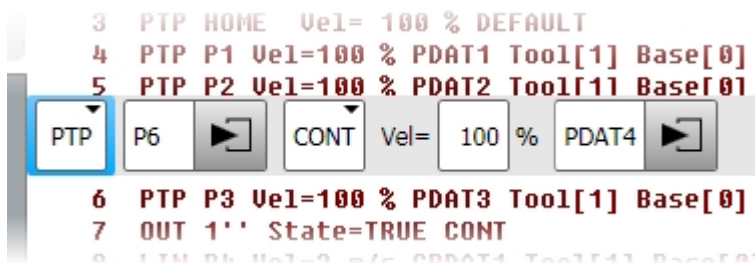


Рис. 10-3: Встроенный формуляр для программирования перемещения

### Виды перемещений

Для программирования команд перемещения доступны различные виды перемещений. В зависимости от требований к рабочему процессу робота можно запрограммировать следующие перемещения:

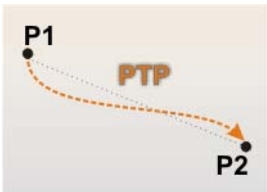
- осевые перемещения (PTP: от точки к точке);
- перемещения по траектории: LIN (линейное) и CIRC (круговое);
- SPLINE: Spline — это перемещение, особенно пригодное для сложных, изогнутых траекторий. Такие траектории можно также создать с помощью перемещений LIN и CIRC, однако по сравнению с ними перемещение Spline обладает определенными преимуществами.

**i** Перемещения Spline не рассматриваются в настоящей учебной документации. Более подробная информация приведена в *Инструкции по эксплуатации и программированию системного программного обеспечения KUKA 8.2.*

### 10.3 Создание перемещений с оптимизацией времени такта (осевое перемещение)

Продолжительность: 00:20:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

#### Вид перемещения PTP

Вид перемещения	Значение	Пример использования
	<p><i>Point-To-Point</i>: от точки к точке</p> <ul style="list-style-type: none"> <li>■ Перемещение по осям: робот проводит точку TSP вдоль <b>самой быстрой</b> траектории к целевой точке. Самой быстрой траекторией, как правило, не является кратчайшая траектория, т. е. <b>не</b> прямая. Так как оси робота совершают вращательное перемещение, криволинейные траектории могут выполняться быстрее, чем прямые.</li> <li>■ Точный ход перемещения непредсказуем.</li> <li>■ Ведущей осью обозначается ось, которой требуется больше всего времени для достижения целевой точки.</li> <li>■ SYNCHRO PTP: все оси запускаются одновременно и также синхронно подходят к останову.</li> <li>■ Первым перемещением в программе должно быть перемещение PTP, поскольку только в нем оцениваются состояние и поворот.</li> </ul>	<p>Точечные приложения, напр.:</p> <ul style="list-style-type: none"> <li>■ точечная сварка;</li> <li>■ транспортировка;</li> <li>■ измерение, контроль.</li> </ul> <p>Вспомогательные положения:</p> <ul style="list-style-type: none"> <li>■ промежуточные точки;</li> <li>■ свободные точки в пространстве.</li> </ul>

#### Synchro PTP

Ведущей осью обозначается ось, которой требуется больше всего времени для достижения целевой точки. При этом учитывается заданная во встроенном формуляре скорость.

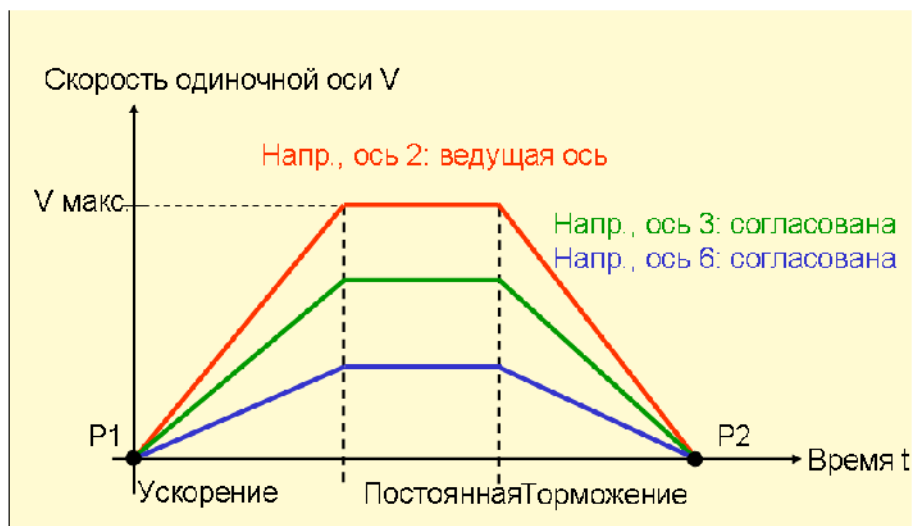


Рис. 10-4: Synchro PTP

**Состояние и поворот**

Состояние и поворот служат для того, чтобы задать однозначное осевое положение из нескольких возможных осевых положений для того же положения точки TCP.

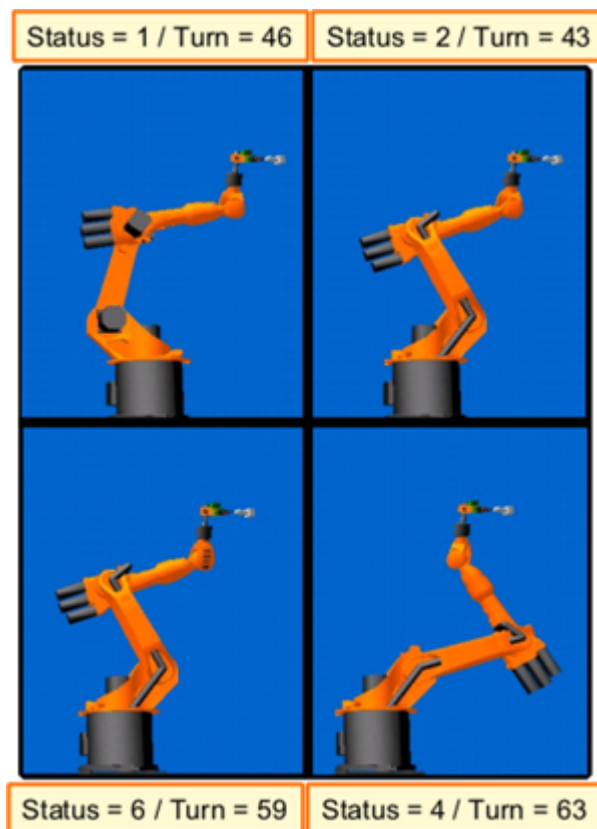


Рис. 10-5: Различные положения осей из-за различных значений Status и Turn

Система управления учитывает программируемые значения состояния и поворота только для перемещений PTP. Перемещения CP игнорируются. Поэтому первая команда перемещения в программе KRL должна быть полной командой PTP типа POS или E6POS, чтобы было определено однозначное положение выходов (или полной командой PTP типа AXIS или E6AXIS).



```

DEFDAT MAINPROGRAM ( )
DECL POS XPOINT1={X 900, Y 0, Z 800, A 0, B 0, C 0, S 6, T 27}
DECL FDAT FPOINT1 ...
...
ENDDDAT

```

## Сглаживание

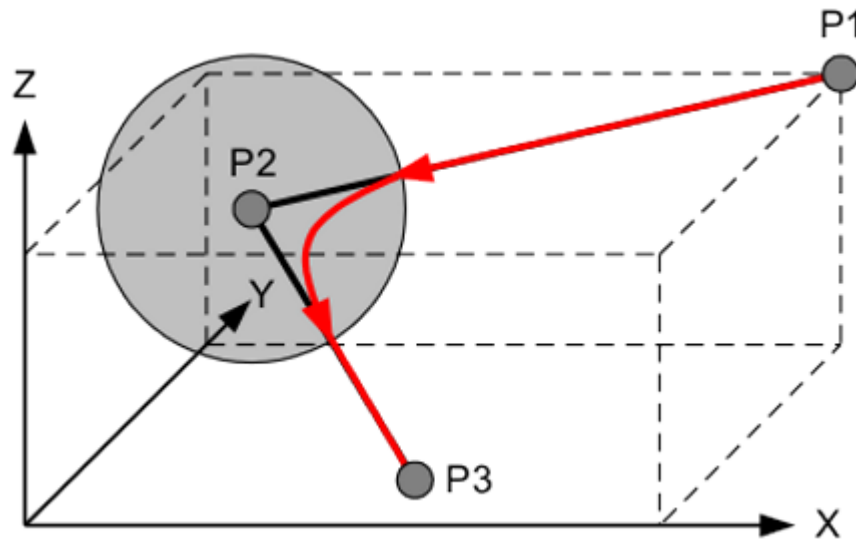


Рис. 10-6: Сглаживание точки

Для ускорения перемещения система управления способна выполнить сглаживание с помощью команд с обозначением CONT. Сглаживание означает, что подвод к координатам точки выполняется неточно. Траектория контура точного останова покидается заранее. Точка TCP проводится вдоль контура сглаживания, который переходит в контур точного останова следующей командой перемещения.

Преимущества сглаживания:

- меньший износ;
- меньшее время такта.

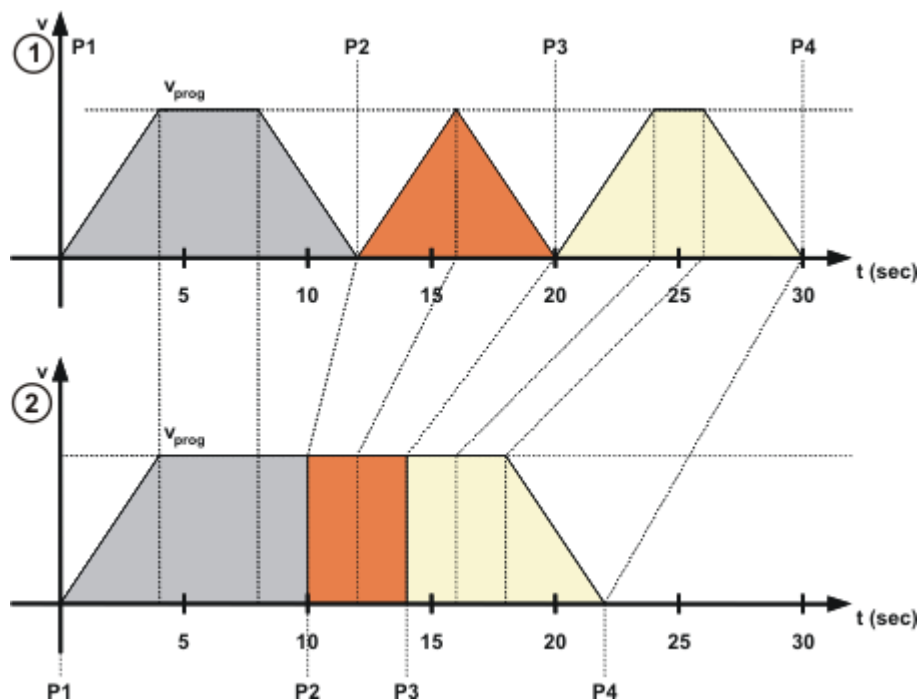
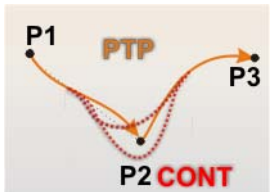


Рис. 10-7: Сглаживание точного останова в сравнении

Чтобы можно было выполнить перемещение со сглаживанием, система управления должна считать следующие записи перемещения. Это выполняется посредством предварительного запуска компьютера.

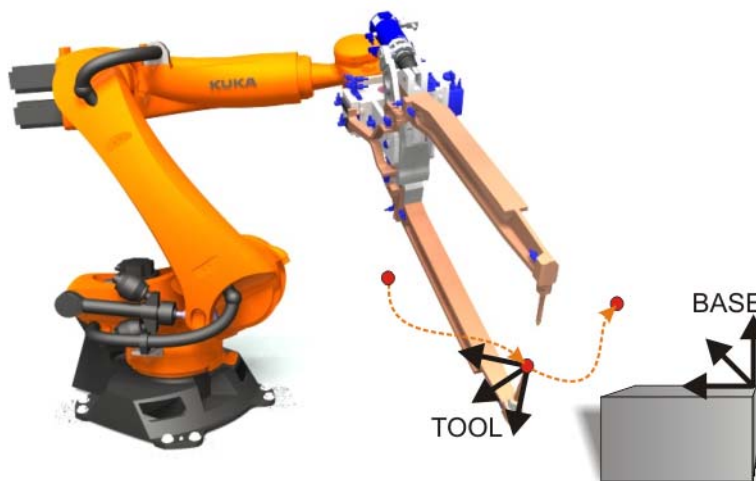
Сглаживание в виде перемещения РТР

Вид перемещения	Особенность	Расстояние сглаживания
	<ul style="list-style-type: none"> <li>Контур сглаживания непредсказуем!</li> </ul>	Заданное процентное значение

**Порядок действий для создания перемещений РТР**

**Необходимые условия**

- Установлен режим работы Т1.
  - Выбрана программа робота.
- Переместить точку TCP в положение, которое должно быть запрограммировано в качестве целевой точки.



**Рис. 10-8: Команда перемещения**

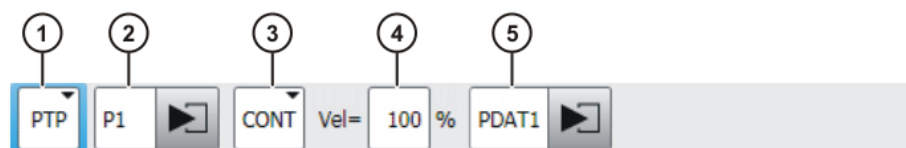
- Установить курсор в строку, после которой должна быть вставлена команда перемещения.
- Выбрать последовательность меню **Команды > Перемещение > РТР**.

В качестве альтернативы в соответствующей строке можно нажать на программируемую клавишу **Перемещение**.

[Показать процедуру на офисном ПК!](#)

Появится встроенный формуляр:

- встроенный формуляр РТР.**



**Рис. 10-9: Встроенный формуляр движения РТР**

- Установить параметры во встроенном формуляре.

По з.	Описание
1	Вид перемещения <b>PTP</b> , <b>LIN</b> или <b>CIRC</b> .
2	Имя целевой точки указывается автоматически, однако его можно перезаписать вручную.  Для обработки данных точки коснуться стрелки; откроется окно опций <b>Фреймы</b> .  Для <b>CIRC</b> помимо целевой точки следует запрограммировать вспомогательную точку: выполнить подвод к положению вспомогательной точки и нажать кнопку <b>Touchup HP</b> .
3	<ul style="list-style-type: none"> <li>■ <b>CONT</b>: сглаживание целевой точки;</li> <li>■ <b>[пусто]</b>: точный подвод в целевую точку.</li> </ul>
4	Скорость: <ul style="list-style-type: none"> <li>■ перемещения PTP: <b>1–100 %</b>;</li> <li>■ перемещения по траектории: <b>0,001–2 м/с</b>.</li> </ul>
5	Набор данных перемещения: <ul style="list-style-type: none"> <li>■ ускорение;</li> <li>■ расстояние сглаживания (при вводе CONT в поле (3));</li> <li>■ контроль ориентации (только для перемещений по траектории).</li> </ul>

5. В окне опций «Фреймы» ввести верные данные для системы координат инструмента и основной системы координат, а также данные о режиме интерполяции (внешняя точка TCP: вкл./выкл.) и контроле возможности столкновения.

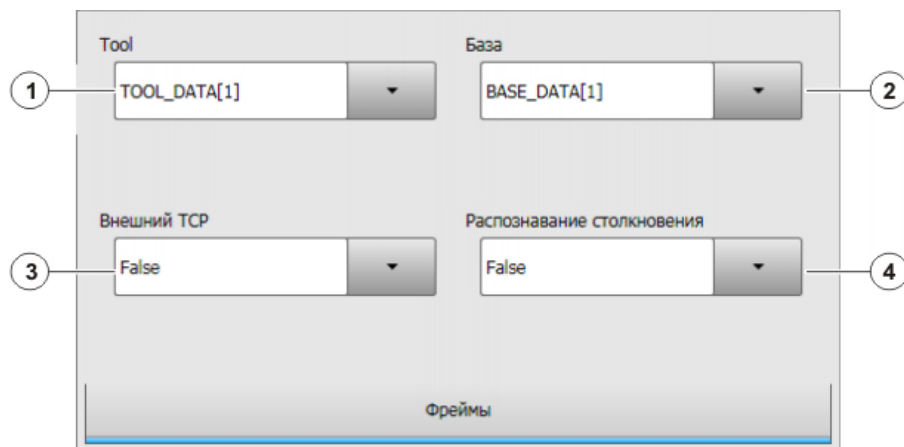


Рис. 10-10: Окно опций Фреймы

Поз.	Описание
1	Выбрать инструмент.  Если установлено <b>True</b> в поле <b>Внешний TCP</b> : выбрать заготовку.  Диапазон значений: [1]–[16].
2	Выбрать базу.  Если установлено <b>True</b> в поле <b>Внешний TCP</b> : выбрать стационарный инструмент.  Диапазон значений: [1]–[32].

Поз.	Описание
3	Режим интерполяции: <ul style="list-style-type: none"> <li>■ <b>False</b>: инструмент смонтирован на установочном фланце;</li> <li>■ <b>True</b>: инструмент является стационарным.</li> </ul>
4	<ul style="list-style-type: none"> <li>■ <b>True</b>: для этого перемещения система управления роботом определяет осевые моменты. Это необходимо для распознавания столкновения;</li> <li>■ <b>False</b>: для этого перемещения система управления роботом не определяет осевые моменты. Поэтому распознавание столкновения для этого перемещения невозможно.</li> </ul>

6. В окне опций «Параметры перемещения» можно уменьшить ускорение с максимального значения. При активации сглаживания можно изменить расстояние сглаживания. В зависимости от конфигурации расстояние устанавливается в мм или %.

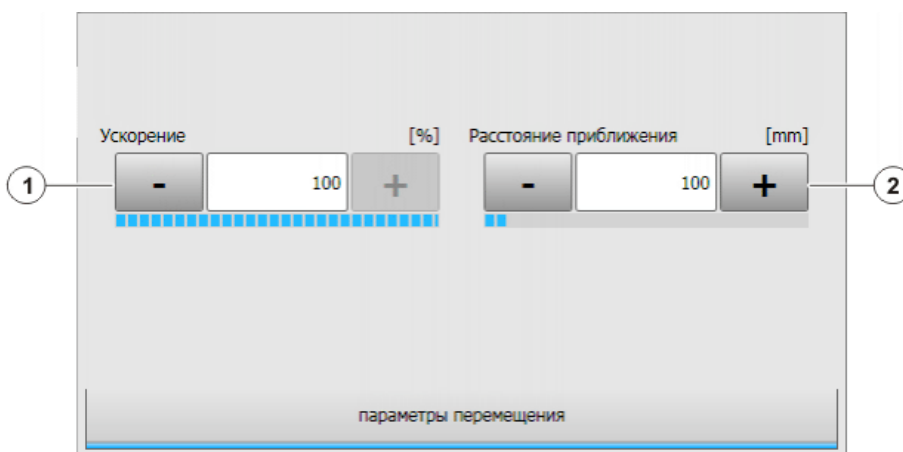


Рис. 10-11: Окно опций параметры перемещения (РТР)

Поз.	Описание
1	Ускорение Относится к указанному в технических данных максимальному значению. Максимальное значение зависит от типа робота и установленного режима работы. Ускорение действует для ведущей оси этой записи перемещений: <ul style="list-style-type: none"> <li>■ <b>1–100 %</b>.</li> </ul>
2	Это поле отображается только при условии, что во встроенном формуляре было выбрано <b>CONT</b> . Расстояние до целевой точки, на которой сглаживание начинается раньше всего. Максимальное расстояние: половина расстояния между начальной и целевой точкой по отношению к контуру перемещения РТР без сглаживания: <ul style="list-style-type: none"> <li>■ <b>1–100 %</b>;</li> <li>■ <b>1–1000 мм</b>.</li> </ul>

7. Сохранить команду, нажав кнопку **Команда ОК**. Текущее положение TCP принимается в качестве целевой точки.

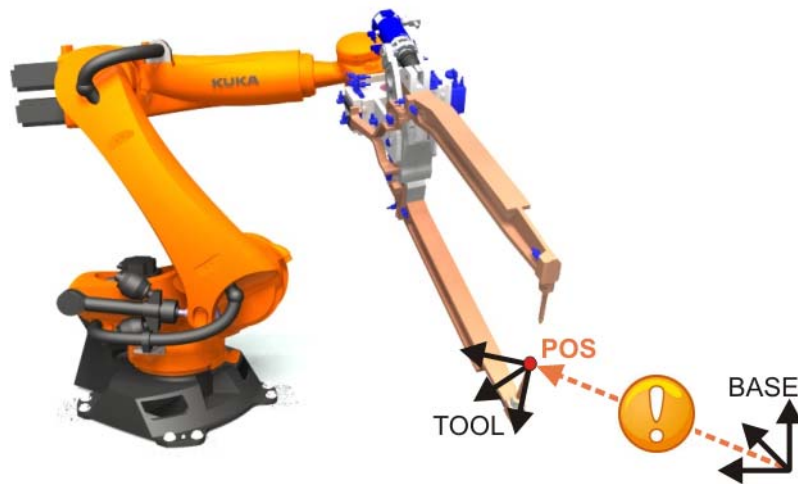


Рис. 10-12: Сохранение координат точки при «Команда ОК» и «Touchup»

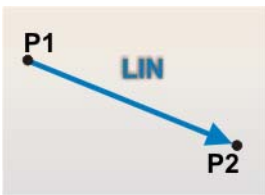
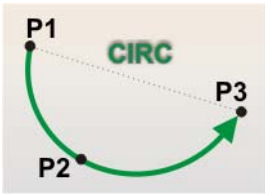
#### 10.4 Создание перемещений по траекториям

---

Продолжительность:	01:00:00
Оснащение:	Офисный ПК
Информация:	
Сертификат:	

---

### Виды перемещения LIN и CIRC

Вид перемещения	Значение	Пример использования
	<p><i>Linear.</i> линейное:</p> <ul style="list-style-type: none"> <li>■ прямолинейное перемещение по траектории;</li> <li>■ точка TCP инструмента перемещается от начальной до заданной точки с постоянной скоростью и заданной ориентацией;</li> <li>■ скорость и ориентация относятся к точке TCP.</li> </ul>	<p>Области применения, связанные с определенной траекторией, напр.:</p> <ul style="list-style-type: none"> <li>■ контурная сварка;</li> <li>■ склеивание;</li> <li>■ лазерная сварка/резка.</li> </ul>
	<p><i>Circular.</i> круговое:</p> <ul style="list-style-type: none"> <li>■ круговая траектория перемещения задается начальной, вспомогательной и целевой точками;</li> <li>■ точка TCP инструмента перемещается от начальной до заданной точки с постоянной скоростью и заданной ориентацией;</li> <li>■ скорость и ориентация относятся к инструменту (системе координат инструмента).</li> </ul>	<p>Области применения, связанные с определенной траекторией, как в случае с перемещением LIN:</p> <ul style="list-style-type: none"> <li>■ круги, радиусы, закругления.</li> </ul>

### Сингулярные положения

Роботы KUKA с 6 степенями свободы имеют 3 различных сингулярных положения.

Степени свободы:

- степень свободы = Degree of Freedom (DOF);
- свободно перемещаемое в пространстве тело имеет 6 степеней свободы:
  - 3 степени свободы для положения в пространстве посредством координат точек X, Y, Z;
  - 3 степени свободы для ориентации в пространстве посредством угла поворота A, B, C;
- благодаря этому робот может перейти в рабочем пространстве в каждую точку в любой ориентации.

Для кинематических систем робота (напр., 6-осевой робот с шарнирными рычагами) существуют точки в пространстве, которые ведут к так называемым **сингулярностям**. Сингулярность характеризуется и определяется тем, что две оси робота являются коллинеарными (находятся на одной прямой). Типичной конфигурацией с сингулярностью является перевернутое положение инструмента. Здесь ось 1 и ось 6 находятся на одной прямой. Система управления не может однозначно присвоить выполняемое вращение вокруг вертикали для оси 1 или оси 6. Другое такое положение возникает при прохождении оси 5 через ноль. Здесь ось 4

и ось 6 располагаются на одной прямой. Здесь существует бесконечное число осевых положений, которые ведут к одинаковому положению инструмента или траекториям перемещения, при которых несколько осей должны были бы двигаться с бесконечной скоростью друг против друга. Некоторые положения прерывают работу программы при прохождении такой точки.

Сингулярное положение характеризуется тем, что однозначная обратная трансформация (пересчет прямоугольных координат в осевые значения), несмотря на то, что заданы состояние и поворот, невозможна. В этом случае, а также если малейшее изменение прямоугольных координат приводит к очень большим изменениям осевых углов, говорят о сингулярных положениях. Это не механическое, а математическое свойство, которое существует по этой причине только в области перемещений по траектории, а не осевых перемещений.

### Сингулярность перевернутых положений $\alpha_1$

При сингулярности перевернутых положений точка запястья (= центральная точка оси A5) лежит вертикально на оси A1 робота.

Положение оси A1 невозможно однозначно определить путем обратной трансформации, поэтому она может принимать любые значения.

Если целевая точка одной записи перемещений PTP лежит в сингулярности перевернутых положений, то система управления роботом может реагировать через системную переменную \$SINGUL\_POS[1] следующим образом:

- 0: угол оси A1 устанавливается на ноль градусов (настройка по умолчанию);
- 1: угол оси A1 остается неизменным от точки запуска до целевой точки.

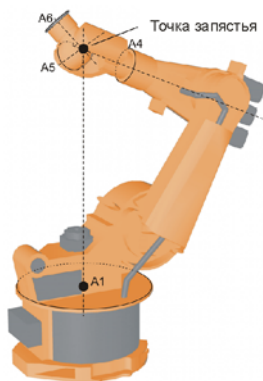


Рис. 10-13: Сингулярность перевернутых положений (положение  $\alpha_1$ )

### Сингулярность развернутых положений $\alpha_2$

При сингулярности развернутых положений точка запястья (= центральная точка оси A5) лежит на продолжении осей A2 и A3 робота.

Робот находится на краю своей рабочей зоны.

Обратной трансформацией можно однозначно определить осевой угол, однако следствием малых скоростей в прямоугольных координатах являются большие осевые скорости осей A2 и A3.

Если целевая точка одной записи перемещений PTP лежит в сингулярности развернутых положений, то система управления роботом может



реагировать через системные переменные  $\$SINGUL\_POS[2]$  следующим образом:

- **0**: угол оси A2 устанавливается на ноль градусов (настройка по умолчанию);
- **1**: угол оси A2 остается неизменным от точки запуска до целевой точки.

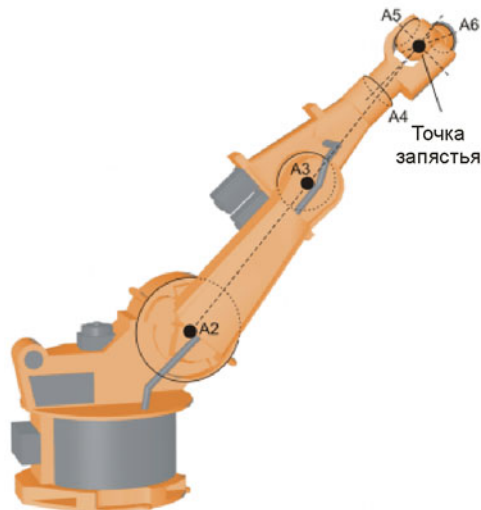


Рис. 10-14: Развернутое положение (положение  $\alpha_2$ )

### Сингулярность ручных осей $\alpha_5$

При сингулярности ручных осей оси A4 и A6 расположены параллельно друг другу, ось A5 в диапазоне  $\pm 0,01812^\circ$ .

Положение обеих осей невозможно однозначно определить путем обратной трансформации. Но существует сколько угодно положений для осей A4 и A6 с одинаковой суммой осевых углов.

Если целевая точка одной записи перемещений PTP лежит в сингулярности этих ручных осей, то система управления роботом может реагировать через системные переменные  $\$SINGUL\_POS[3]$  следующим образом:

- **0**: угол оси A4 устанавливается на ноль градусов (настройка по умолчанию);
- **1**: угол оси A4 остается неизменным от начальной точки до целевой точки.

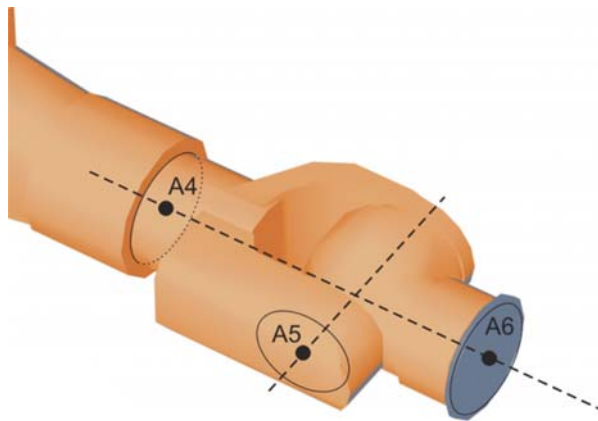


Рис. 10-15: Сингулярность ручных осей (положение  $\alpha_5$ )

### Контроль ориентации при перемещениях по траектории

При перемещениях по траектории можно точно определить контроль ориентации. Инструмент может иметь различные ориентации в начальной и в целевой точке перемещения.

Контроль ориентации для вида перемещения **LIN**:

- **Стандартный или Ручной PTP;**

Ориентация инструмента во время перемещения постоянно меняется.

Использовать значение «Ручной PTP» в том случае, если робот попадает в сингулярность ручных осей со значением «Стандартный», поскольку ориентирование осуществляется посредством линейного перевода углов ручной оси (осевое перемещение).

**Ручной PTP** не подходит для случаев, когда следует остановить определенный процесс ориентирования, напр., лазерную сварку.

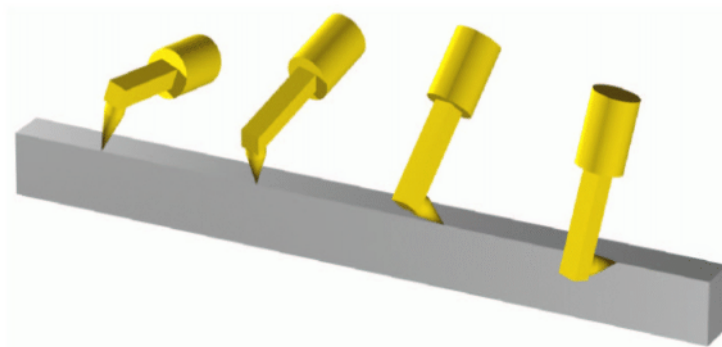
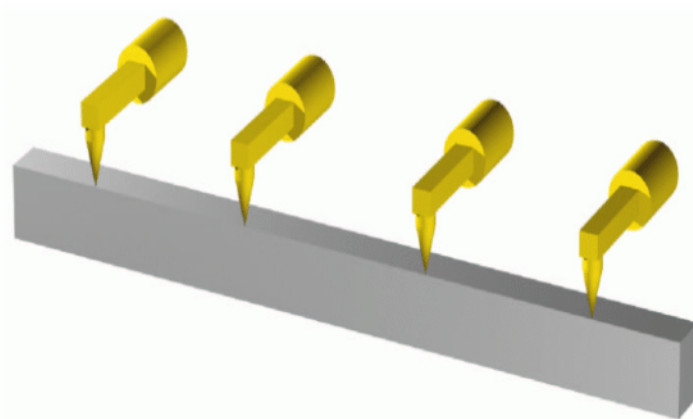


Рис. 10-16: Стандартный

- **Постоянный.**

Ориентация инструмента во время перемещения остается постоянной, как было запрограммировано в начальной точке. Запрограммированная в конечной точке ориентация игнорируется.



**Рис. 10-17: Постоянная ориентация**

Контроль ориентации для вида перемещения **CIRC**:

- **Стандартный или Ручной РТР;**

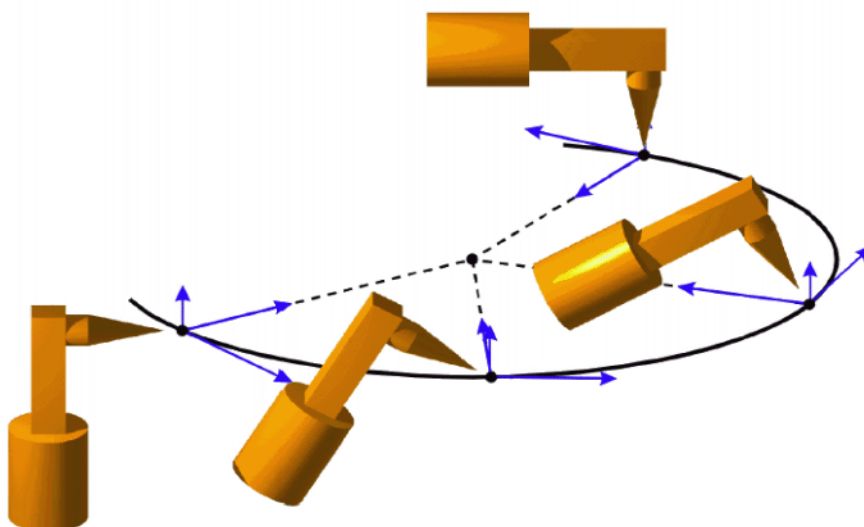
Ориентация инструмента во время перемещения постоянно меняется.

Использовать значение «Ручной РТР» в том случае, если робот попадает в сингулярность ручных осей со значением «Стандартный», поскольку ориентирование осуществляется посредством линейного перевода углов ручной оси (осевое перемещение).

---

**Ручной РТР** не подходит для случаев, когда следует остановить определенный процесс ориентирования, напр., лазерную сварку.

---



**Рис. 10-18: Стандарт + относительно базы**

- **Постоянный.**

Ориентация инструмента во время перемещения остается постоянной, как было запрограммировано в начальной точке. Запрограммированная в конечной точке ориентация игнорируется.

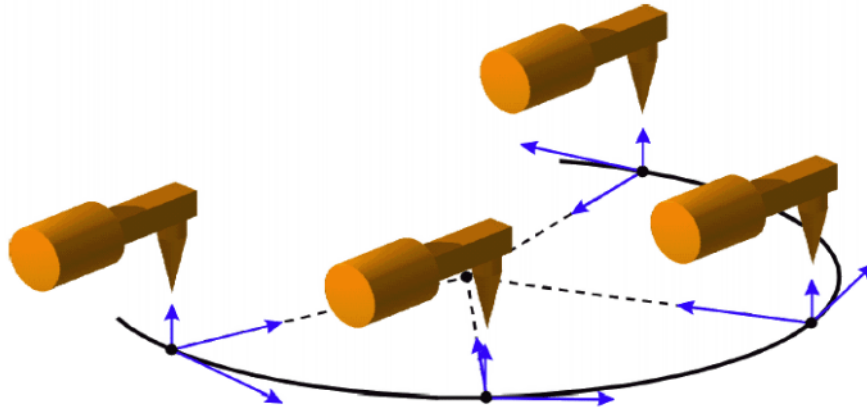


Рис. 10-19: Постоянная ориентация + относительно базы

### Последовательность перемещений при CIRC

Здесь отсчетная точка инструмента или заготовки перемещается по дуге окружности к целевой точке. Траектория описывается начальной, вспомогательной и конечной точками. При этом начальной точкой является целевая точка предыдущей команды перемещения.

Ориентация инструмента во вспомогательной точке не играет никакой роли.

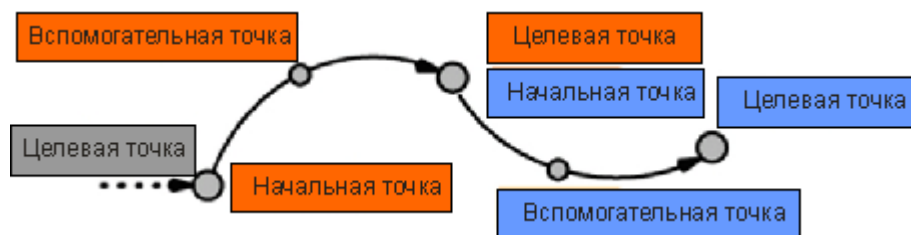


Рис. 10-20: Два круговых сегмента с CIRC

### Сглаживание перемещений по траектории



Функция сглаживания не предназначена для создания круговых перемещений. Речь идет исключительно о предотвращении точного останова в точке.

#### Недостатки сглаживания:

- Траектория задается сглаживаемыми точками, которые лежат не на траектории. Зоны сглаживания труднопредсказуемы. Создать желаемую траекторию сложно.
- зачастую возникают труднопредсказуемые уменьшения скорости, например, в зонах сглаживания и близлежащих точках;
- ход траектории изменяется, если сглаживание невозможно;
- ход траектории изменяется в зависимости от коррекции, скорости или ускорения.

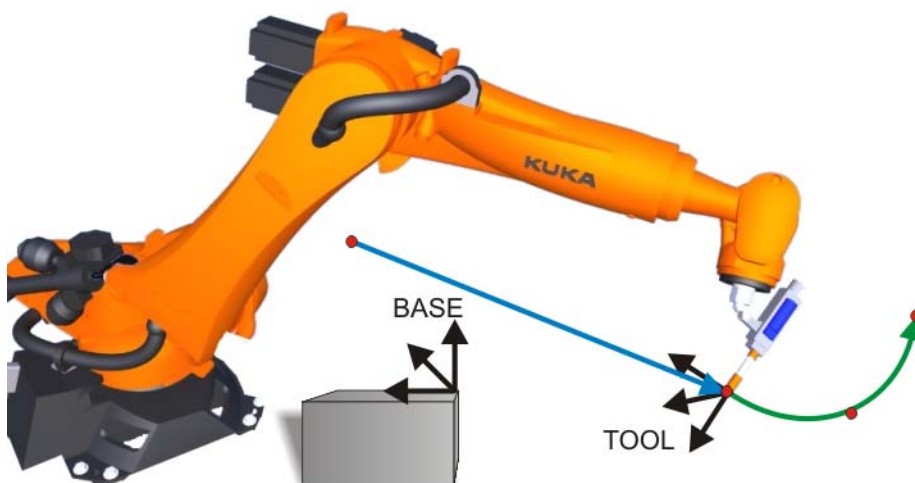
Сглаживание в видах перемещений PTP, LIN и CIRC

Вид перемещения	Особенность	Расстояние сглаживания
	<ul style="list-style-type: none"> <li>Ход траектории соответствует двум параболическим нагрузкам</li> </ul>	Указание значений в мм
	<ul style="list-style-type: none"> <li>Ход траектории соответствует двум параболическим нагрузкам</li> </ul>	Указание значений в мм

**Порядок действий для создания перемещений LIN и CIRC**

**Необходимые условия**

- Установлен режим работы T1.
  - Выбрана программа робота.
- Переместить точку TCP в положение, которое должно быть запрограммировано в качестве целевой точки.



**Рис. 10-21: Команда перемещения с LIN и CIRC**

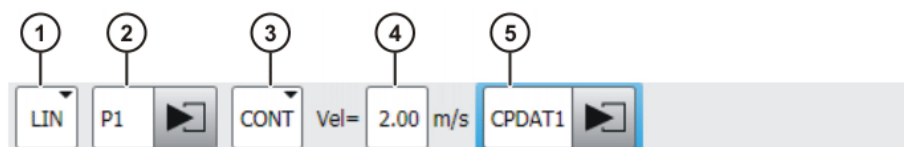
- Установить курсор в строку, после которой должна быть вставлена команда перемещения.
- Выбрать последовательность меню **Команды > Перемещение > LIN** или **CIRC**.

В качестве альтернативы в соответствующей строке можно нажать на программируемую клавишу **Перемещение**.

[Показать процедуру на офисном ПК!](#)

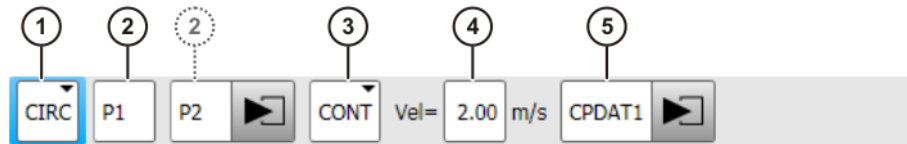
Появится встроенный формуляр:

- встроенный формуляр LIN;**



**Рис. 10-22: Встроенный формуляр движения LIN**

■ **встроенный формуляр CIRC.**

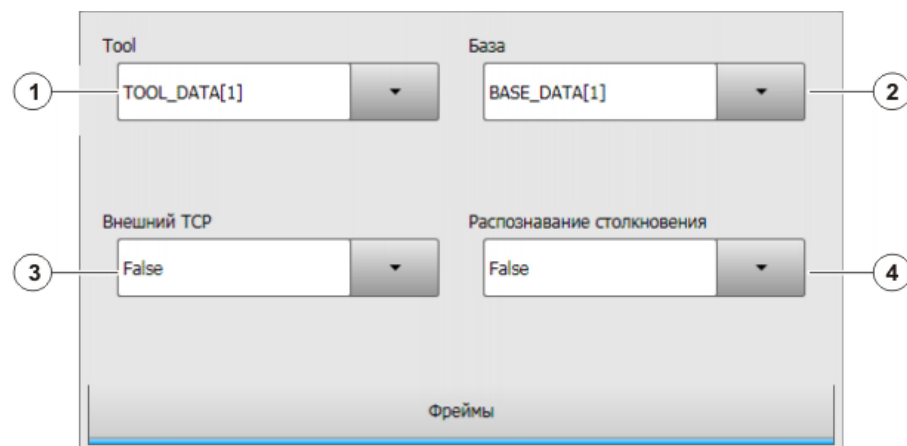


**Рис. 10-23: Встроенный формуляр перемещения CIRC**

4. Установить параметры во встроенном формуляре.

По з.	Описание
1	Вид перемещения <b>PTP</b> , <b>LIN</b> или <b>CIRC</b> .
2	Имя целевой точки указывается автоматически, однако его можно перезаписать вручную.  Для обработки данных точки коснуться стрелки; откроется окно опций <b>Фреймы</b> .  Для <b>CIRC</b> помимо целевой точки следует запрограммировать вспомогательную точку: выполнить подвод к положению вспомогательной точки и нажать кнопку <b>Touchup HP</b> . Ориентация инструмента во вспомогательной точке не играет никакой роли.
3	<ul style="list-style-type: none"> <li>■ <b>CONT</b>: сглаживание целевой точки;</li> <li>■ <b>[пусто]</b>: точный подвод в целевую точку.</li> </ul>
4	Скорость: <ul style="list-style-type: none"> <li>■ перемещения PTP: <b>1–100 %</b>;</li> <li>■ перемещения по траектории: <b>0,001–2 м/с</b>.</li> </ul>
5	Набор данных перемещения: <ul style="list-style-type: none"> <li>■ ускорение;</li> <li>■ расстояние сглаживания (при вводе CONT в поле (3));</li> <li>■ контроль ориентации (только для перемещений по траектории).</li> </ul>

5. В окне опций «Фреймы» ввести верные данные для системы координат инструмента и основной системы координат, а также данные о режиме интерполяции (внешняя точка TCP: вкл./выкл.) и контроле возможности столкновения.



**Рис. 10-24: Окно опций Фреймы**

Поз.	Описание
1	<p>Выбрать инструмент.</p> <p>Если установлено <b>True</b> в поле <b>Внешний TCP</b>: выбрать заготовку.</p> <p>Диапазон значений: [1]–[16].</p>
2	<p>Выбрать базу.</p> <p>Если установлено <b>True</b> в поле <b>Внешний TCP</b>: выбрать стационарный инструмент.</p> <p>Диапазон значений: [1]–[32].</p>
3	<p>Режим интерполяции:</p> <ul style="list-style-type: none"> <li>■ <b>False</b>: инструмент смонтирован на установочном фланце;</li> <li>■ <b>True</b>: инструмент является стационарным.</li> </ul>
4	<ul style="list-style-type: none"> <li>■ <b>True</b>: для этого перемещения система управления роботом определяет осевые моменты. Это необходимо для распознавания столкновения;</li> <li>■ <b>False</b>: для этого перемещения система управления роботом не определяет осевые моменты. Поэтому распознавание столкновения для этого перемещения невозможно.</li> </ul>

6. В окне опций «Параметры перемещения» можно уменьшить ускорение с максимального значения. При активации сглаживания можно изменить расстояние сглаживания. В дальнейшем также можно изменить контроль ориентации.

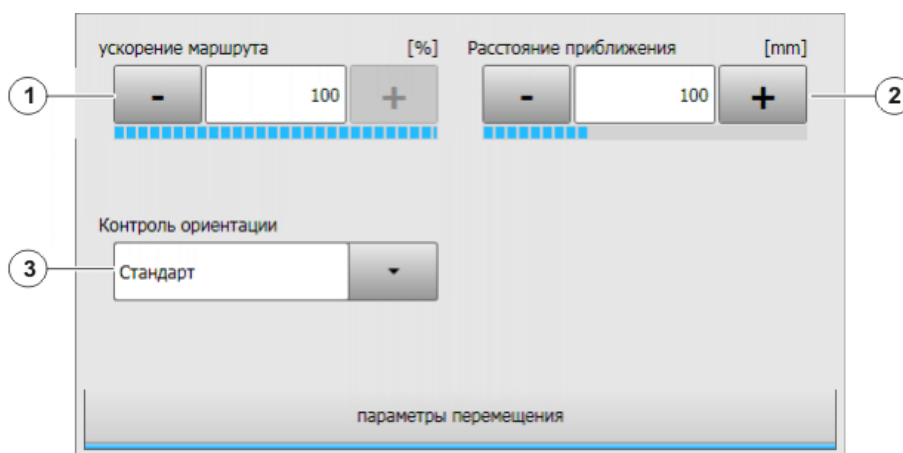


Рис. 10-25: Окно опций параметры перемещения (LIN, CIRC)

Установка ускорения используется, напр., при транспортировке жидкостей или работе с большими вакуумными захватами.



Поз.	Описание
1	Ускорение  Относится к указанному в технических данных максимальному значению. Максимальное значение зависит от типа робота и установленного режима работы.
2	Расстояние до целевой точки, на которой сглаживание начинается раньше всего.  Это расстояние не должно превышать половину расстояния между начальной и целевой точкой. Если здесь будет введено большее значение, оно игнорируется, и используется максимальное значение.  Это поле отображается только при условии, что во встроенном формуляре было выбрано <b>CONT</b> .
3	Выбрать контроль ориентации: <ul style="list-style-type: none"><li>■ стандартный;</li><li>■ ручной РТР;</li><li>■ постоянный контроль ориентации.</li></ul> (>>> "Контроль ориентации при перемещениях по траектории" Страница 149)

7. Сохранить команду, нажав кнопку **Команда ОК**. Текущее положение ТСП принимается в качестве целевой точки.

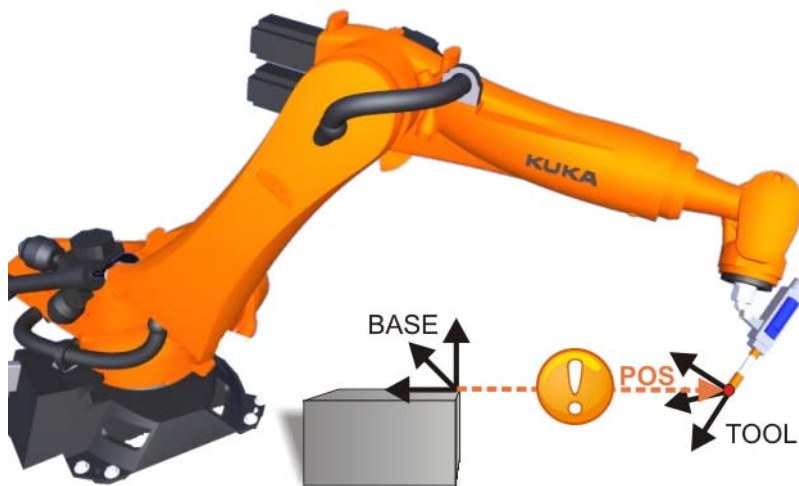


Рис. 10-26: Сохранение координат точки при «Команда ОК» и «Touchup»

## 10.5 Изменение команд перемещения

---

Продолжительность: 00:30:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

---

### Изменение команд перемещения

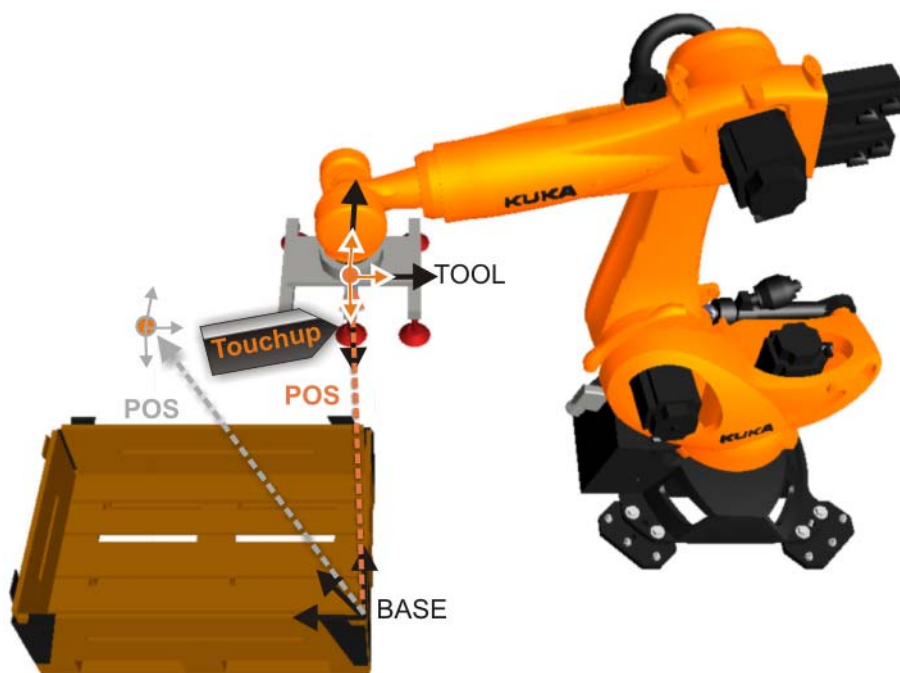
Существуют различные причины для изменения имеющихся команд перемещения:

Примеры причин	Выполняемое изменение
Изменяется положение захватываемой детали. Изменяется положение одного из пяти отверстий при обработке. Следует укоротить сварной шов.	Изменение данных положения
Изменяется положение палеты.	Изменение данных фрейма: база и/или инструмент
Положение было ошибочно запрограммировано с неверной базой.	Изменение данных фрейма: база и/или инструмент с обновлением положения
Обработка выполняется слишком медленно: следует улучшить время такта.	Изменение данных перемещения: скорость, ускорение Изменение вида перемещения

**Эффекты при изменении команд перемещения**

**Изменение данных положения**

- Изменяется лишь набор данных точки: точка получает новые координаты, поскольку с помощью Touchup значения обновляются. Старые координаты точки перезаписываются и после этого более недоступны.



**Рис. 10-27: Изменение положения робота с помощью Touchup**

**Изменение данных фреймов**

- При изменении данных фреймов (напр., инструмент, база) возникает смещение положения (ср. с векторным смещением).
- Положение робота изменяется. Старые координаты точки по-прежнему сохраняются и являются действительными. Изменяется только точка отсчета (напр., база).
- Может возникнуть выход за пределы рабочей области! Поэтому определенные положения робота являются недостижимыми.
- Если положение робота должно остаться прежним, но требуется изменить параметры фрейма, то после изменения параметров (напр.,

базы) в нужном положении следует обновить координату с помощью Touchup.

**i** Диалоговое окно пользователя также предупреждает: «Внимание! При изменении связанных с точкой параметров фрейма существует опасность столкновения!».

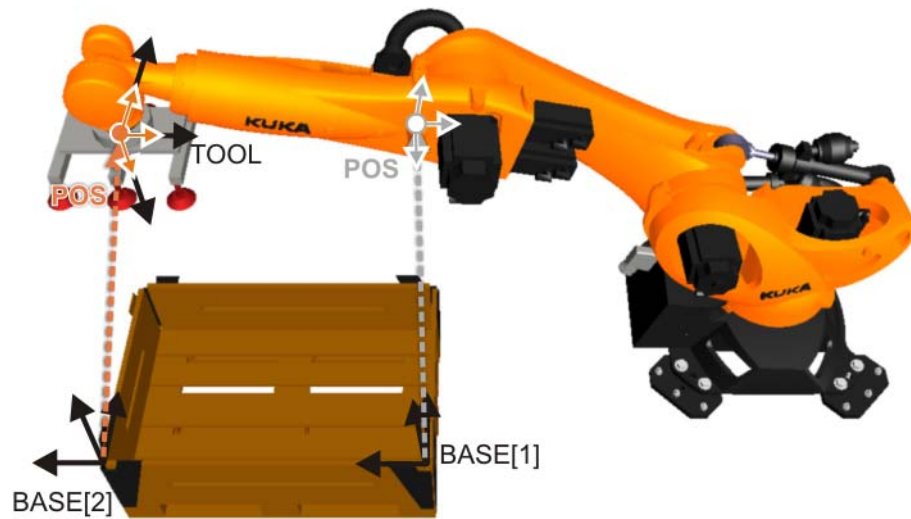


Рис. 10-28: Изменение данных фрейма (пример базы)

#### Изменение данных перемещения

- При изменении скорости или ускорения изменяется профиль перемещения. Это может оказать влияние на процесс обработки, особенно в областях применения, связанных с определенной траекторией:
  - толщина клеевого валика;
  - качество сварного шва.

#### Изменение вида перемещения

- Изменение вида перемещения всегда ведет к изменению планирования траектории. Это может привести в неблагоприятных случаях к столкновениям, поскольку траектория может непредсказуемо измениться.

Использование: при проблемах с изменением состояния при рассчитанных положениях можно обойти проблему, переключившись с LIN на PTP.

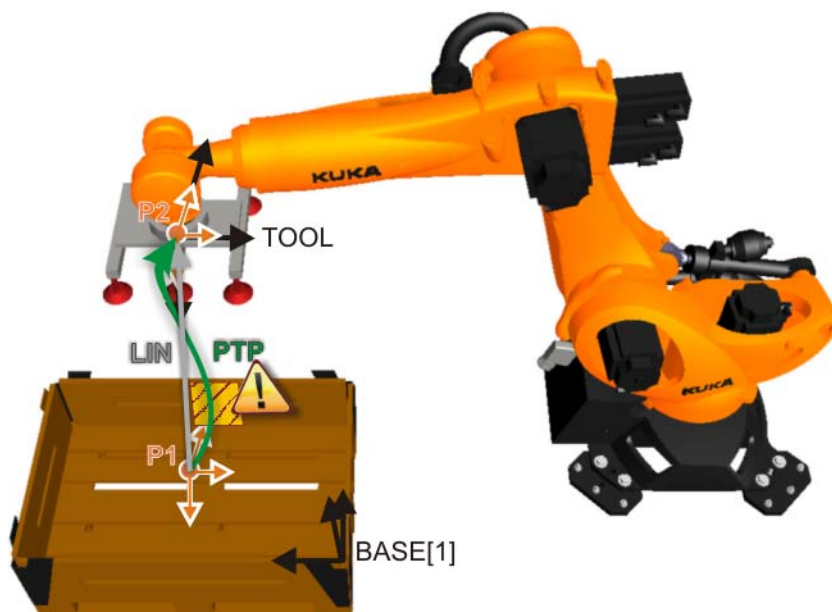


Рис. 10-29: Изменение вида перемещения

Указания по технике безопасности для изменения команд перемещения

**УВЕДОМЛЕНИЕ**

После каждого изменения команд перемещения следует протестировать программу робота при пониженной скорости (режим работы T1).

Немедленный запуск программы робота при высокой скорости может привести к повреждению системы робота и всей установки, так как возможны непредсказуемые перемещения. Если в опасной зоне находится какое-либо лицо, возможны опасные для жизни травмы.

Изменение параметров перемещения – фреймы

1. Установить курсор в строку с командой, которую следует изменить.
2. Нажать кнопку **Изменить**. Откроется встроенный формуляр команды.
3. Открыть окно опций «Фреймы».
4. Установить новый инструмент или базу или внешнюю точку TSP.
5. Подтвердить пользовательское диалоговое окно «Внимание! При изменении параметров фрейма существует опасность столкновения!» нажатием на кнопку **ОК**.
6. Если требуется **сохранить текущее положение робота** с измененными настройками инструмента или базы, следует удерживать кнопку **Touch Up**, чтобы повторно рассчитать и сохранить текущее положение.
7. Сохранить изменения, нажав кнопку **Команда ОК**.



При изменении параметров фрейма следует повторно проверить программы на предмет отсутствия столкновений.

Изменение положения

[Показать процедуру на офисном ПК](#)

Порядок действий для изменения положения робота:

1. Выбрать режим работы T1 и установить курсор в строку с командой, которую следует изменить.
2. Привести робот в нужное положение.
3. Нажать кнопку **Изменить**. Откроется встроенный формуляр команды.

4. Для перемещений PTP и LIN:
  - нажать кнопку **Touchup**, чтобы принять текущее положение TCP в качестве новой целевой точки.
- Для перемещений CIRC:
  - нажать кнопку **Touchup HP**, чтобы принять текущее положение TCP в качестве новой вспомогательной точки;
  - или нажать кнопку **Touchup ZP**, чтобы принять текущее положение TCP в качестве новой целевой точки.
5. Подтвердить операцию, ответив **Да** на контрольный запрос.
6. Сохранить изменения, нажав кнопку **Команда ОК**.

---

В качестве альтернативы для **PTP** и **LIN** можно непосредственно обновить точку строки, в которой стоит курсор, посредством кнопки **Touchup**. Для **CIRC** при аналогичном порядке действий следует предварительно ответить на запрос, какую точку следует обновить: HP или ZP.

---

### Изменение параметров перемещения

Этот порядок действий можно использовать для внесения следующих изменений:

- Вид перемещения
  - Скорость
  - Ускорение
  - Сглаживание
  - Расстояние сглаживания
1. Установить курсор в строку с командой, которую следует изменить.
  2. Нажать кнопку **Изменить**. Открывается встроенный формуляр команды.
  3. Изменить параметры.
  4. Сохранить изменения, нажав кнопку **Команда ОК**.



При изменении параметров перемещения следует повторно проверить программы на предмет отсутствия столкновений и безопасности процесса.



## 11 Использование логических функций в программе робота

### 11.1 Обзор

В этом учебном разделе рассматриваются следующие элементы:

- программирование функций ожидания;
- программирование простых функций переключения;
- программирование функций переключения траектории.



Рис. 11-1: Смена главы

### 11.2 Введение в программирование логики

---

Продолжительность: 00:15:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

---

Использование входов и выходов при программировании логики

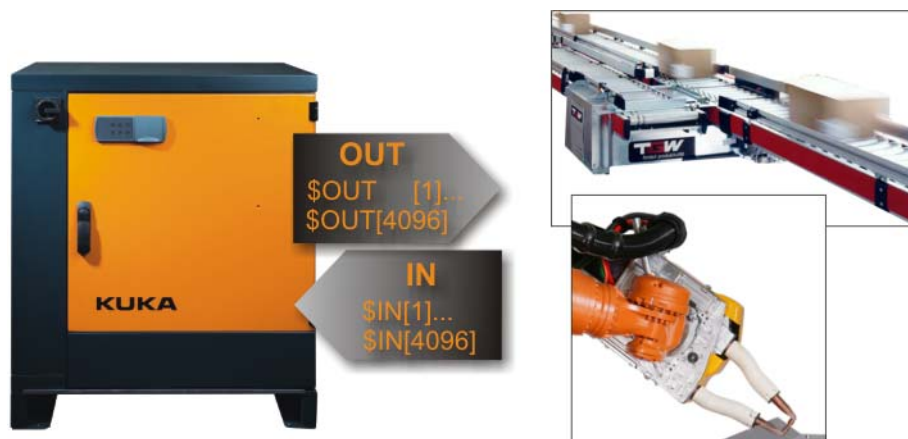


Рис. 11-2: Цифровые входы и выходы

Чтобы обеспечить связь системы управления роботом с периферией, можно использовать цифровые и аналоговые входы и выходы.

Определение терминов



Термин	Определение	Пример
<b>Связь</b>	Обмен сигналами по интерфейсу	Запрос состояния (захват открыт/ закрыт)
<b>Периферия</b>	«Окружение»	Инструмент (напр., захват, сварочные клещи и т. д.), датчики, системы транспортировки материала и т. д.
<b>Цифровой</b>	Цифровая техника: Дискретные по значению и времени сигналы	Сигнал датчика: наличие детали: значение 1 (TRUE/ истинно), отсутствие детали: значение 0 (FALSE/ложно)
<b>Аналоговый</b>	Изображение физической величины	Измерение температуры
<b>Входы</b>	<i>Поступающие</i> к системе управления через интерфейс магистральной шины сигналы	Сигнал датчика: захват открыт/захват закрыт
<b>Выходы</b>	<i>Отправляемые</i> от системы управления через интерфейс магистральной шины сигналы	Команда переключения клапана, которая ведет к закрыванию крюка для захвата.

При программировании роботов KUKA применяются входные и выходные сигналы для логических команд:

- **OUT**: переключение выхода в определенном месте программы;
- **WAIT FOR**: зависящая от сигнала функция ожидания: здесь система управления ожидает сигнал:
  - вход **IN**;
  - выход **OUT**;
  - сигнал времени **TIMER**;
  - внутренний адрес памяти системы управления (маркер/однорядная память) **FLAG** или **CYCFLAG** (при постоянной циклической оценке);
- **WAIT**: зависящая от времени функция ожидания: система управления ожидает в этом месте программы в течение введенного времени.

### 11.3 Программирование функций ожидания

---

Продолжительность: 00:30:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

---

#### Предварительный запуск компьютера

При предварительном запуске компьютера считываются записи перемещения (невидимые для оператора) во время предварительного выполнения, чтобы система управления могла планировать траекторию при

командах сглаживания. С предварительным выполнением обрабатываются не только данные перемещения, но и арифметические и управляемые периферией команды.

```

1 DEF Depal_Box1( )
2
3  INI
4  PTP HOME Vel= 100 % DEFAULT
5  PTP P1 Vel=100 % PDAT1 Tool[5]:GRP1 Base[10]:STAT1
6  PTP P2 Vel=100 % PDAT2 Tool[5]:GRP1 Base[10]:STAT1
7  LIN P3 Vel=1 m/s CPDAT1 Tool[5]:GRP1 Base[10]:STAT1
8  LIN P4 Vel=1 m/s CPDAT2 Tool[5]:GRP1 Base[10]:STAT1
9  PTP P5 Vel=100 % PDAT3 Tool[5]:GRP1 Base[10]:STAT1
10 OUT 26'' State=TRUE
11 PTP HOME Vel=100 % PDAT4
12
13  FND
  
```

Рис. 11-3: Предварительный запуск компьютера

Строка	
6	Положение указателя основного выполнения (серая полоса)
9	Возможное положение указателя предварительного выполнения (не видно)
10	Набор команд, который приводит к срабатыванию предварительного выполнения

Некоторые команды приводят к срабатыванию останова предварительного выполнения. К ним относятся команды, влияющие на периферию, напр., команды OUT (закрывать захват, открыть сварочные клещи). Если указатель предварительного выполнения остановится, то сглаживание невозможно.

## Функции ожидания

Функции ожидания в программе перемещения позволяют просто выполнить программирование с помощью встроенных формуляров. Различают зависимую от времени функцию ожидания и зависимую от сигнала функцию ожидания.

С помощью **WAIT** перемещение робота останавливается на запрограммированное время. WAIT всегда приводит к останову предварительного выполнения.



Рис. 11-4: Встроенный формуляр WAIT

Поз.	Описание
1	Время ожидания ■ $\geq 0$ с

### Пример программы:

```

PTP P1 Vel=100% PDAT1
PTP P2 Vel=100% PDAT2
WAIT Time=2 sec
PTP P3 Vel=100% PDAT3
  
```

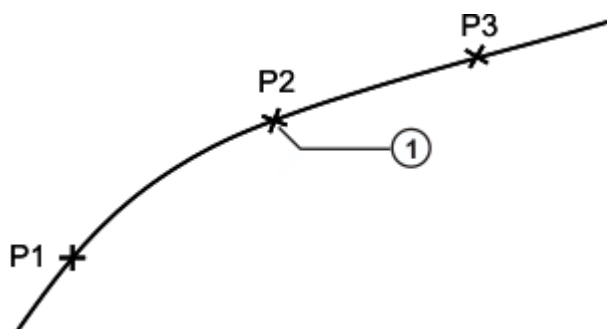


Рис. 11-5: Пример перемещения для логики

Поз.	Примечание
1	Перемещение прерывается на 2 секунды в точке P2.

**WAIT FOR** устанавливает зависимость от сигнала функцию ожидания.

При необходимости можно логически связать несколько сигналов (максимум 12). Если добавляется создание связи, то во встроенном формуляре добавляются поля для дополнительных сигналов и для других связей.

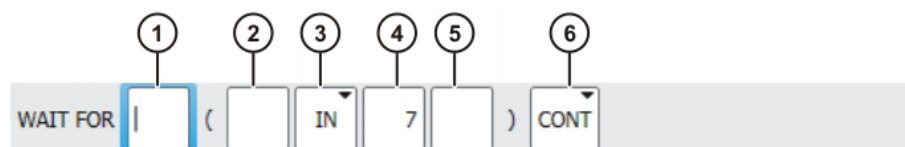


Рис. 11-6: Встроенный формуляр WAITFOR

Поз.	Описание
1	<p>Добавить внешнюю связь. Оператор находится между заключенными в скобки выражениями:</p> <ul style="list-style-type: none"> <li>■ <b>AND;</b></li> <li>■ <b>OR;</b></li> <li>■ <b>EXOR.</b></li> </ul> <p>Добавить NOT.</p> <ul style="list-style-type: none"> <li>■ <b>NOT;</b></li> <li>■ <b>[пусто].</b></li> </ul> <p>Ввести нужный оператор с помощью соответствующей экранной кнопки.</p>
2	<p>Добавить внутреннюю связь. Оператор находится внутри заключенного в скобки выражения:</p> <ul style="list-style-type: none"> <li>■ <b>AND;</b></li> <li>■ <b>OR;</b></li> <li>■ <b>EXOR.</b></li> </ul> <p>Добавить NOT.</p> <ul style="list-style-type: none"> <li>■ <b>NOT;</b></li> <li>■ <b>[пусто].</b></li> </ul> <p>Ввести нужный оператор с помощью соответствующей экранной кнопки.</p>

Поз.	Описание
3	Ожидаемый сигнал: <ul style="list-style-type: none"> <li>■ IN;</li> <li>■ OUT</li> <li>■ CYCFLAG;</li> <li>■ TIMER;</li> <li>■ FLAG.</li> </ul>
4	Номер сигнала <ul style="list-style-type: none"> <li>■ 1 - 4096</li> </ul>
5	Если у сигнала имеется имя, это имя отображается. Только для группы пользователей «Эксперты»: нажатием на кнопку <b>Описание</b> можно ввести имя. Имя задается свободно.
6	<ul style="list-style-type: none"> <li>■ <b>CONT</b>: обработка в предварительном выполнении</li> <li>■ <b>[пусто]</b>: обработка с остановом предварительного выполнения</li> </ul>



При использовании записи CONT следует учитывать, что сигнал будет опрошен в предварительном выполнении. Изменение сигнала после времени предварительного выполнения не определяется.

## Логические связи

Не путать с логическими сравнениями! В случае с логическими сравнениями операторами являются «больше», «меньше/равно» и т. д.

Логические связи также используются в зависящих от сигналов функциях ожидания. С помощью логических связей можно сочетать запрос различных сигналов или состояний: напр., можно создать зависимости, а также исключить определенные состояния.

Результат функции с логическим оператором всегда содержит истинное значение, т. е. в конце всегда находится значение TRUE (значение 1) или значение FALSE (значение 0).



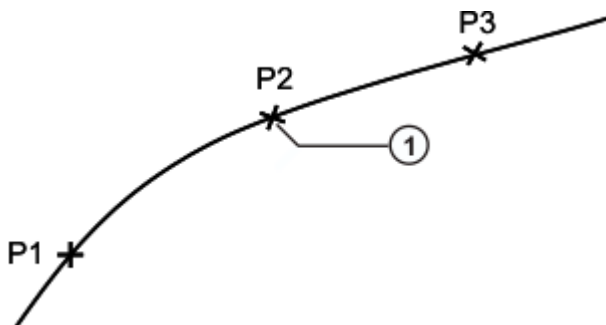
Рис. 11-7: Пример и принцип логической связи

Операторы для логических связей:

- **NOT**: этот оператор используется для отрицания, т. е. значение изменяется на противоположное (TRUE превращается в FALSE);
- **AND**: результат этого выражения является истинным, если оба связанных выражения являются истинными;
- **OR**: результат этого выражения является истинным, если по крайней мере одно из связанных выражений является истинным;
- **EXOR**: результат этого выражения является истинным, если оба связанных оператором выражения имеют разные значения истинности.

**Обработка с предварительным выполнением и без него (CONT)**

Зависящие от сигнала функции ожидания можно запрограммировать как с обработкой в предварительном выполнении, так и без нее. **Без предварительного выполнения** означает, что перемещение остановится в точке в любом случае, а сигнал будет проверен в ней: (1) (>>> Рис. 11-8 ) (1). Таким образом, сглаживание этой точки невозможно.

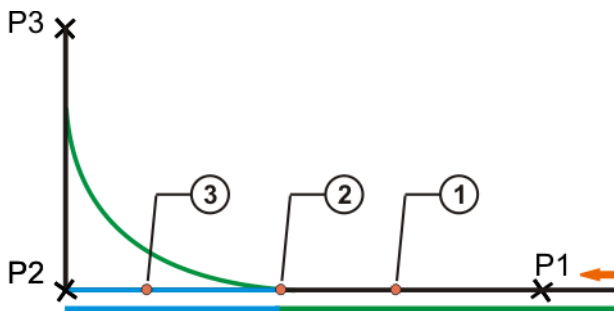


**Рис. 11-8: Пример перемещения для логики**

```
PTP P1 Vel=100% PDAT1
PTP P2 CONT Vel=100% PDAT2
WAIT FOR IN 10 'door_signal'
PTP P3 Vel=100% PDAT3
```

**i** При обработке строки с WAIT FOR без CONT появится информационное сообщение: «Сглаживание невозможно».

Зависящие от сигнала функции ожидания, запрограммированные с **предварительным выполнением**, позволяют выполнить сглаживание точки, созданной перед строкой команды. Однако текущее положение указателя предварительного выполнения неоднозначно (стандартное значение: три записи перемещений), поэтому точный момент времени для проверки сигнала не определен (>>> Рис. 11-9 ) (1). Кроме того, изменения сигнала после проверки сигнала не определяются.



**Рис. 11-9: Пример перемещения для логики с предварительным выполнением**

```
PTP P1 Vel=100% PDAT1
PTP P2 CONT Vel=100% PDAT2
WAIT FOR IN 10 'door_signal' CONT
PTP P3 Vel=100% PDAT3
```

**Порядок действий**

1. Установить курсор в строку, после которой должна быть вставлена логическая команда.
2. Выбрать последовательность меню **Команды > Логика > WAIT FOR** или **WAIT**.
3. Во встроенном формуляре установить параметры.
4. Сохранить команду, нажав кнопку **Команда ОК**.

## 11.4 Программирование простых функций переключения

Продолжи-  
тельность: 00:20:00  
Оснащение: Офисный ПК  
Информация:  
Сертификат:

### Простая функция переключения

С помощью функции переключения можно отправить цифровой сигнал к периферии. Для этого используется номер выхода, который следует заранее задать в интерфейсе соответствующим образом.

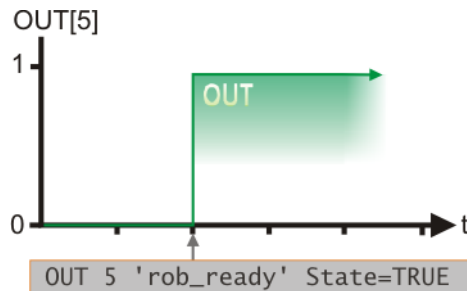


Рис. 11-10: Статическое переключение

Сигнал устанавливается статично, т. е. он сохраняется до тех пор, пока для выхода не будет задано другое значение. Функция переключения реализуется в программе с помощью встроенного формуляра:

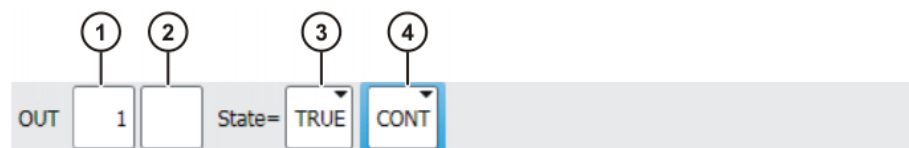


Рис. 11-11: Встроенный формуляр OUT

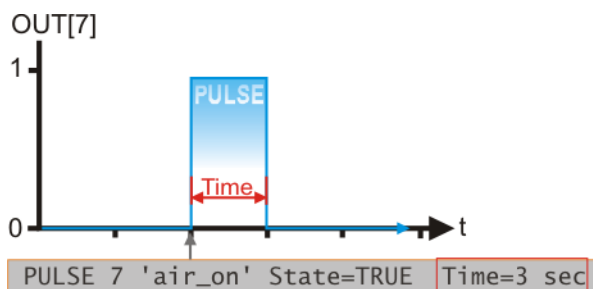
Поз.	Описание
1	Номер выхода <ul style="list-style-type: none"> <li>1 ... 4096</li> </ul>
2	Если у выхода есть имя, это имя отображается. Только для группы пользователей «Эксперты»: нажатием на кнопку <b>Описание</b> можно ввести имя. Имя задается свободно.
3	Состояние, в которое переключается выход <ul style="list-style-type: none"> <li>TRUE</li> <li>FALSE.</li> </ul>
4	<ul style="list-style-type: none"> <li>CONT: обработка в предварительном выполнении</li> <li>[пусто]: обработка с остановом предварительного выполнения</li> </ul>



При использовании записи CONT следует учитывать, что сигнал будет установлен в предварительном выполнении.

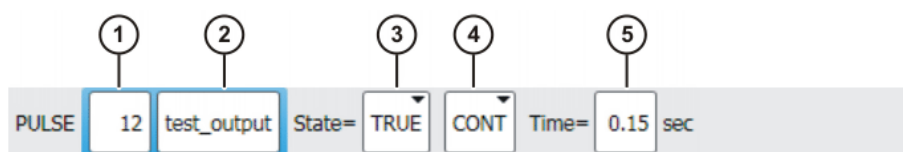
**Импульсные функции переключения**

Как и в случае с простой функцией переключения, здесь изменяется значение выхода. Конечно, в случае с импульсом сигнал принимается обратно через определенное время.



**Рис. 11-12: Импульсный уровень**

Программирование осуществляется с помощью встроенного формуляра, который устанавливает импульс с определенной длиной.



**Рис. 11-13: Встроенный формуляр PULSE**

Поз.	Описание
1	Номер выхода: ■ 1–4096.
2	Если у выхода есть имя, это имя отображается. Только для группы пользователей «Эксперты»: нажатием на кнопку <b>Описание</b> можно ввести имя. Имя задается свободно.
3	Состояние, в которое переключается выход: ■ <b>TRUE</b> : уровень «High»; ■ <b>FALSE</b> : уровень «Low».
4	■ <b>CONT</b> : обработка в предварительном выполнении; ■ <b>[пусто]</b> : обработка с остановом предварительного выполнения.
5	Длина импульса: ■ 0,10–3,00 с.

**Действие записи CONT с функциями переключения**

При опускании записи CONT во встроенном формуляре OUT в процессе переключения принудительно осуществляется **останов предварительного выполнения**, а в точке перед командой переключения выполняется точный останов. После установки выхода перемещение продолжается.

```

LIN P1 Vel=0.2 m/s CPDAT1
LIN P2 CONT Vel=0.2 m/s CPDAT2
LIN P3 CONT Vel=0.2 m/s CPDAT3
OUT 5 'rob_ready' State=TRUE
LIN P4 Vel=0.2 m/s CPDAT4
    
```



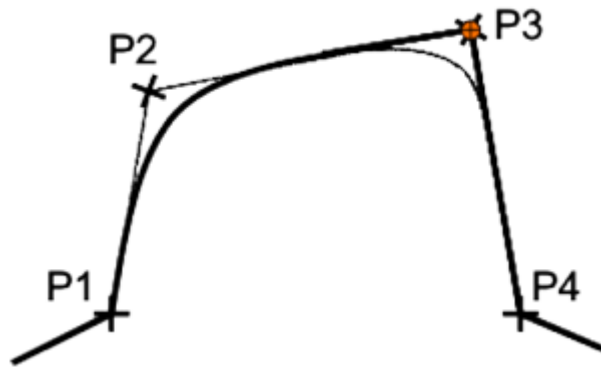


Рис. 11-14: Пример перемещения с переключением с остановом предварительного выполнения

Установка записи CONT приводит к тому, что указатель предварительного выполнения не останавливается (не срабатывает останов предварительного выполнения). Так можно выполнить сглаживание перемещения перед командой переключения. Установка сигнала осуществляется в предварительном выполнении.

```

LIN P1 Vel=0.2 m/s CPDAT1
LIN P2 CONT Vel=0.2 m/s CPDAT2
LIN P3 CONT Vel=0.2 m/s CPDAT3
OUT 5 'rob_ready' State=TRUE CONT
LIN P4 Vel=0.2 m/s CPDAT4

```

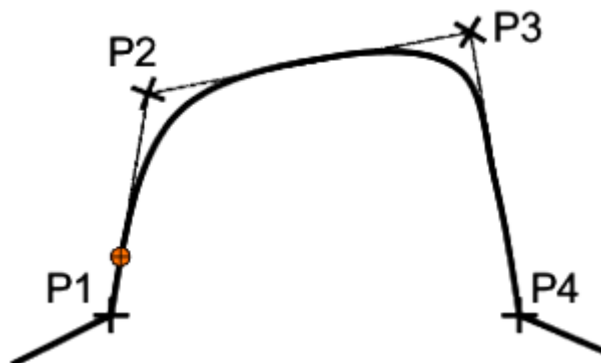


Рис. 11-15: Пример перемещения с переключением в предварительном выполнении

**i** Стандартное значение для указателя предварительного выполнения составляет три строки. Однако предварительное выполнение может меняться от случая к случаю, т. е. следует учитывать, что момент переключения не всегда одинаков.

#### Порядок действий

1. Установить курсор в строку, после которой должна быть вставлена логическая команда.
2. Выбрать последовательность меню **Команды > Логика > OUT > OUT** или **PULSE**.
3. Во встроенном формуляре установить параметры.
4. Сохранить команду, нажав кнопку **Команда ОК**.

## 11.5 Программирование функций переключения траектории

Продолжительность: 00:30:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

### Общая информация

Функцию переключения траектории можно использовать для того, чтобы установить выход в определенной точке на траектории, не прерывая перемещение робота. Различают статическое (SYN OUT) и динамическое (SYN Pulse) переключение. В случае с переключением SYN OUT речь идет о том же сигнале, что и в случае с SYN PULSE. Различаются лишь вид и способ переключения.

### Опция Path

С помощью опции Path можно запустить операцию переключения относительно целевой точки записи перемещения. Операция переключения может быть смещена в пространстве и/или во времени. Запись отсчетного перемещения может представлять собой перемещение LIN или CIRC. Перемещение PTP **недопустимо**.

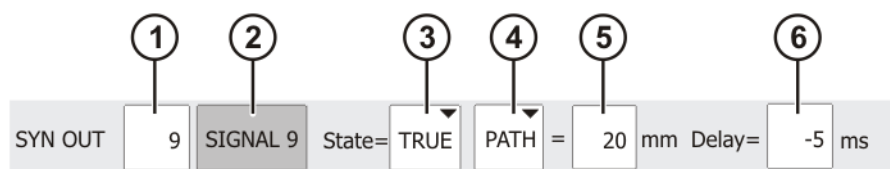


Рис. 11-16: Встроенный формуляр SYN OUT, опция Path

Поз.	Описание	Диапазон значений
1	Номер выхода:	1–4096.
2	Если у выхода есть имя, это имя отображается.  Только для группы пользователей «Эксперты»: нажатием на программируемую клавишу <b>Описание</b> можно ввести имя.	Выбирается свободно
3	Состояние, в которое переключается выход.	TRUE, FALSE
4	Точка, в которой происходит переключение:  ■ <b>PATH</b> : переключение происходит относительно целевой точки записи перемещения.	START, END Опция PATH:

Поз.	Описание	Диапазон значений
5	Пространственное смещение операции переключения. <b>Указание:</b> указание места привязано к целевой точке записи перемещения. Положение точки переключения не меняется несмотря на изменение скорости робота.	-2000 – +2000 мм
6	Временное смещение операции переключения. <b>Указание:</b> временное смещение относится к пространственному смещению.	-1000 – +1000 мс

### Действие опции переключения Path

Пример программы:

фрезер должен переключиться на траектории. Через 20 мм после точки P2 следует на весу начать обработку детали. Чтобы фрезер через 20 мм (путь = 20) после точки P2 достиг полного числа оборотов, его следует включить заранее за 5 мс (задержка = -5 мс).

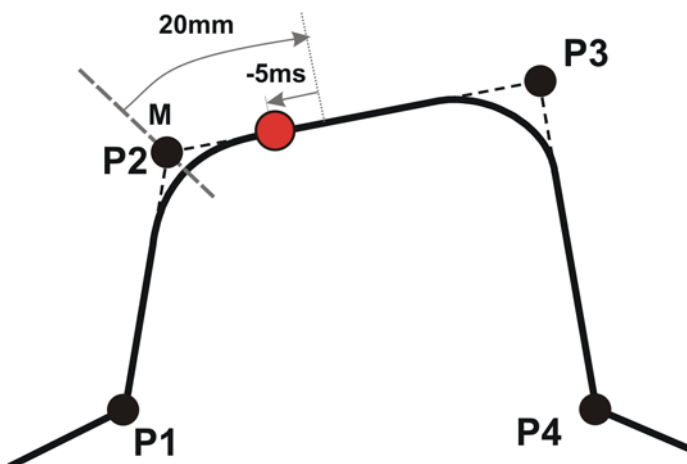


Рис. 11-17: Точка переключения SYN OUT Path

```

LIN P1 VEL=0.3m/s CPDAT1
;Функция переключения по отношению к P2
SYN OUT 9 'SIGNAL 9' Status= True Path=20 Delay=-5ms
LIN P2 CONT VEL=0.3m/s CPDAT2
LIN P3 CONT VEL=0.3m/s CPDAT3
LIN P4 VEL=0.3m/s CPDAT4

```

Границы переключения

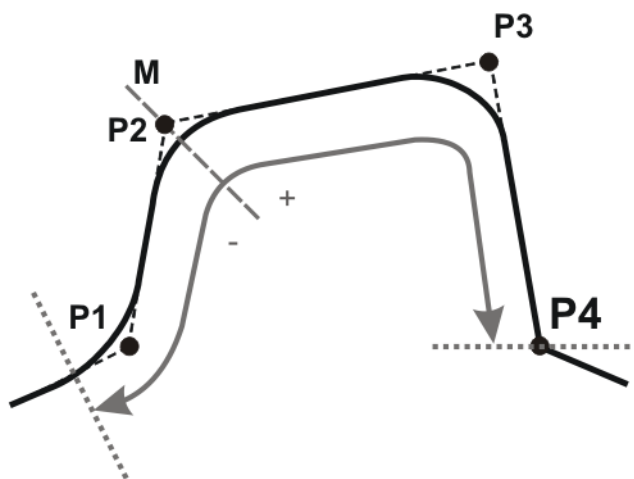


Рис. 11-18: Границы переключения SYN OUT PATH

**Опция Start/End**

Операция переключения может быть запущена относительно начальной или целевой точки записи перемещения. Операция переключения может быть смещена **во времени**. Запись отсчетного перемещения может представлять собой перемещение LIN, CIRC или PTP.

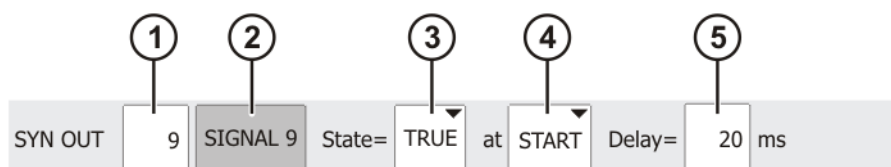


Рис. 11-19: Встроенный формуляр SYN OUT, опция START

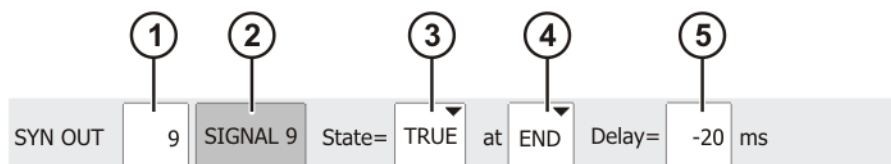


Рис. 11-20: Встроенный формуляр SYN OUT, опция END

Поз.	Описание	Диапазон значений
1	Номер выхода:	1–4096.
2	Если у выхода есть имя, это имя отображается. Только для группы пользователей «Эксперты»: нажатием на программируемую клавишу <b>Описание</b> можно ввести имя.	Выбирается свободно
3	Состояние, в которое переключается выход.	TRUE, FALSE

Поз.	Описание	Диапазон значений
4	<p>Точка, в которой происходит переключение:</p> <ul style="list-style-type: none"> <li>■ <b>START</b>: переключение происходит относительно начальной точки записи перемещения.</li> <li>■ <b>END</b>: переключение происходит относительно целевой точки записи перемещения.</li> </ul>	<p>START, END</p> <p>Опция PATH:</p>
5	<p>Временное смещение операции переключения.</p> <p><b>Указание:</b> заданное время является абсолютным. Положение точки переключения изменяется в зависимости от скорости робота.</p>	-1000 – +1000 мс

#### Порядок действий

1. Установить курсор в строку, после которой должна быть вставлена логическая команда.
2. Выбрать последовательность меню **Команды > Логика > OUT > SYN OUT** или **SYN PULSE**.
3. Во встроенном формуляре установить параметры.
4. Сохранить команду, нажав кнопку **Команда ОК**.



## 12 Введение в уровень эксперта

### 12.1 Обзор

В этом учебном разделе рассматриваются следующие элементы:

- Использование уровня эксперта



Рис. 12-1: Смена главы

### 12.2 Использование уровня эксперта

#### Описание

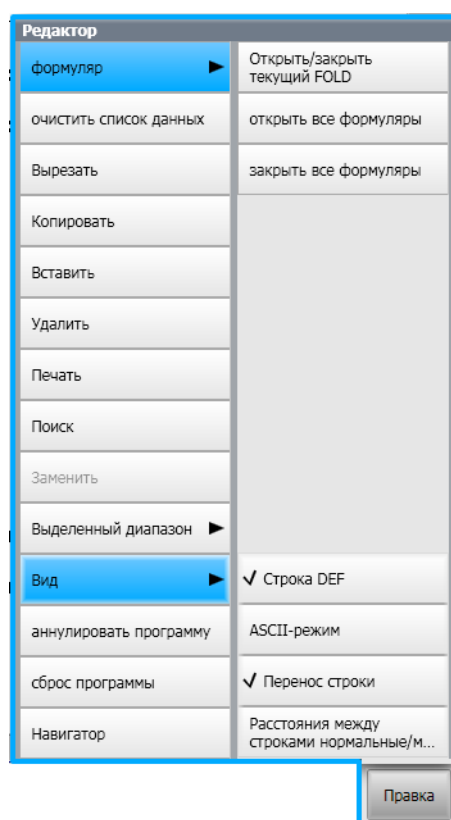
Система управления роботом предусматривает несколько **групп пользователей** с различными функциями. Можно выбрать следующие **группы пользователей**:

- **Операторы**  
Группа пользователей для операторов. Это группа пользователей по умолчанию.
- **Пользователи**  
Группа пользователей для операторов. (Группы пользователей «Операторы» и «Пользователи» предназначены по умолчанию для одной и той же целевой группы).
- **Эксперты**  
Группа пользователей для программистов. Эта группа пользователей защищена паролем.
- **Администраторы**  
Функции соответствуют группе пользователей «Эксперты». Кроме того, возможна интеграция дополнительных программных модулей в систему управления роботом. Эта группа пользователей защищена паролем.
- **Специалист по техобслуживанию систем безопасности**  
Эти пользователи с помощью кода активации могут активировать существующую конфигурацию системы безопасности робота. Если в системе управления роботом не используется опция безопасности, например KUKA.SafeOperation или KUKA.SafeRangeMonitoring, специалист по техобслуживанию систем безопасности имеет дополнительные права. В этом случае он, например, имеет право конфигурировать стандартные функции безопасности.
- **Специалист по вводу систем безопасности в эксплуатацию**  
Эта группа пользователей требуется только в том случае, если используются программы KUKA.SafeOperation и KUKA.SafeRangeMonitoring. Группа пользователей защищена паролем.



**Расширенные функции группы пользователей «Эксперты»:**

- Защита паролем (по умолчанию: kuka).
- Возможно программирование в редакторе с помощью KRL
- Доступен детальный вид модулей
- Отображение/скрытие строки DEF
- Открывание и закрывание формуляров (FOLD)
- Отображение детального вида в программе
- При создании программы можно выбрать один из предварительно заданных шаблонов
- Выход из группы пользователей «Эксперт» выполняется автоматически,
  - если режим работы переключается на AUT или AUT EXT.
  - если в течение определенного времени на панели управления не производится никаких действий (300 сек.).

**Рис. 12-2: Меню «Обработать»**

**i** Для группы пользователей «Эксперт» в меню **Редактировать** доступны все функции.

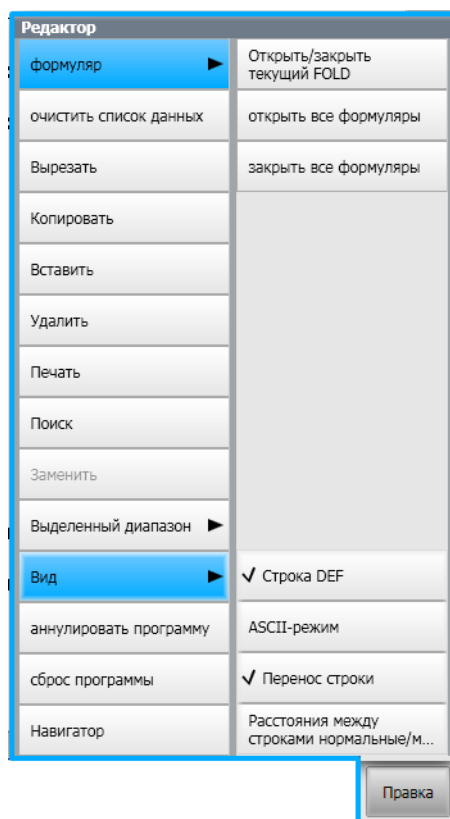
**Функции****Создание программ с помощью шаблонов**

- **Cell**: имеющаяся программа Cell может быть только заменена или создана заново при удаленном Cell.
- **Expert**: модуль, состоящий из файла SRC и DAT, в котором есть только шапка и конец программы.
- **Expert Submit**: дополнительный файл Submit (SUB), состоящий из шапки и конца программы.
- **Function**: создание функции SRC, при котором создается только шапка функции с переменной BOOL. Конец функции имеется, но необходимо запрограммировать возврат.

- **Modul:** модуль, состоящий из файла SRC и DAT, в котором есть шапка и конец программы, а также основная структура (INI и 2x PTP HOME).
- **Submit:** дополнительный файл Submit (SUB), состоящий из шапки и конца программы, а также основной структуры (DECLARATION, INI, LOOP/ENDLOOP).

**Фильтр** задает, как программы должны отображаться в списке файлов. Для выбора имеются указанные ниже **фильтры**.

- Аннулирование или закрытие программы.
- Нажать кнопку **Обработать > Фильтр** и затем **Деталь** или **Модуль**.
- **Деталь**  
Программы отображаются в виде файлов SRC и DAT. (настройка по умолчанию)
- **Модули**  
Программы отображаются как модули.



**Рис. 12-3: Меню «Обработать»**

#### Открытие/закрытие **ФОРМУЛЯРА**

- **ФОРМУЛЯРЫ** для пользователя всегда закрыты и могут открываться группой пользователей «Эксперт».
- Эксперт может также программировать собственные **ФОРМУЛЯРЫ**.
- **Синтаксис** для формуляра:

```
; FOLD имя
Команды
; ENDFOLD <Имя>
```

Строки ENDFOLD легче интерпретировать, если здесь также указать имя формуляра. Формуляры можно вкладывать.

### Отображение/скрытие строки DEF

- **Строка DEF** скрыта по умолчанию. Объявления в программе возможны только, если **строка DEF** отображена.
- **Строка DEF** отдельно отображается и скрывается при каждом открытии и выборе программ. Если включен детальный вид, то **строка DEF** видна и отображать ее отдельно не требуется.

### Порядок действий для активации уровня эксперта и устранения ошибок

#### Активация уровня эксперта

1. В главном меню выберите пункт **Конфигурация > Группа пользователей**.
2. Зарегистрируйтесь как **Эксперт**: Нажмите кнопку **Зарегистрироваться**. Выделите группу пользователей **Эксперт** и нажмите кнопку **Зарегистрироваться** для подтверждения.
3. Введите требуемый пароль (по умолчанию: kuka) и нажмите кнопку **Зарегистрироваться** для подтверждения.

#### Устранение ошибок в программе

1. Выберите дефектный модуль в навигаторе



Рис. 12-4: Программа с ошибками

2. Выберите меню **Список ошибок**
3. Откроется индикация ошибок (*programmname.ERR*)
4. Выберите ошибку, внизу в окне индикации отобразится подробное описание
5. В окне индикации ошибок нажмите кнопку **Показать** и перейдите к содержащей ошибку программе
6. Устраните ошибку
7. Выйдите из редактора и сохраните

## 13 Циклы, обусловленные команды и различение ситуаций

### 13.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Циклы
- Обусловленные команды
- Различение ситуаций



Рис. 13-1: Смена главы

### 13.2 Контроль выполнения программы

---

Продолжительность:	00:30:00
Оснащение:	Офисный ПК
Информация:	
Сертификат:	

---

После рассмотрения команд перемещения, переключения и ожидания участники должны научиться считывать элементы управления выполнением программы в программах робота: циклы, переходы и т. д. Речь не идет о самостоятельном программировании, а лишь о понимании.

#### Контроль выполнения программы

Помимо простых команд перемещения и команд связи (функции переключения и ожидания) в программах робота находится большое количество процедур, которые служат для контроля выполнения программы. К ним относятся:

- **циклы:** циклы – контрольные структуры. Они повторяют блок команд до тех пор, пока не будет выполнено условие прерывания:
  - бесконечные циклы;
  - циклы с подсчетом;
  - отклоняемые и неотклоняемые циклы;
- **переходы:** с помощью переходов можно назначить выполнение разделов программы только при определенном условии:
  - условные переходы;
  - многократные переходы.

### 13.3 Циклы

Продолжительность: 00:30:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

#### Бесконечный цикл

Бесконечный цикл повторяет блок команд бесконечно часто. Цикл можно покинуть с помощью преждевременного прерывания (с помощью функции **EXIT**).

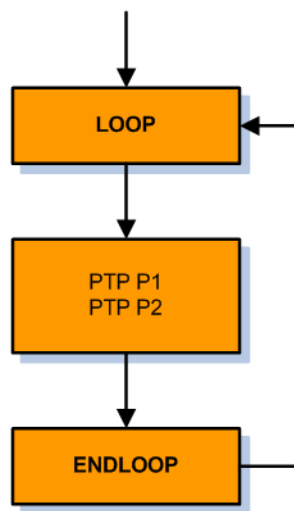


Рис. 13-2: План выполнения программы: бесконечный цикл

Примеры команды LOOP:

■ без **EXIT**:

команды перемещения в точки P1 и P2 выполняются в течение длительного времени;

```
LOOP
  PTP P1 Vel=100% PDAT1
  PTP P2 Vel=100% PDAT2
ENDLOOP
```

■ с **EXIT**:

команды перемещения в точки P1 и P2 выполняются до того, пока вход 30 не переключится на значение TRUE.

```
LOOP
  PTP P1 Vel=100% PDAT1
  PTP P2 Vel=100% PDAT2
  IF $IN[30]==TRUE THEN
    EXIT
  ENDIF
ENDLOOP
```

#### Цикл с подсчетом

С помощью цикла с подсчетом (цикла FOR) можно повторить команду определенное число раз. Количество проходов управляется числовой переменной.

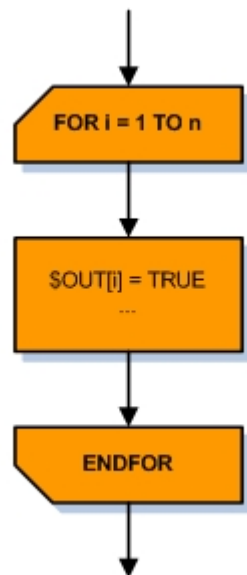


Рис. 13-3: План выполнения программы: цикл FOR

Пример цикла FOR: выходы 1–5 переключаются друг за другом на значение TRUE. Для подсчета проходов внутри цикла используется целочисленная (целая) переменная «i».

```

INT i
...
FOR i=1 TO 5
  $OUT[i] = TRUE
ENDFOR
  
```

#### Отклоняемый цикл

Цикл WHILE – *отклоняемый цикл* или цикл с *предварительной проверкой*, при котором проверяется условие прерывания до того, как будет выполнена часть цикла с командами.

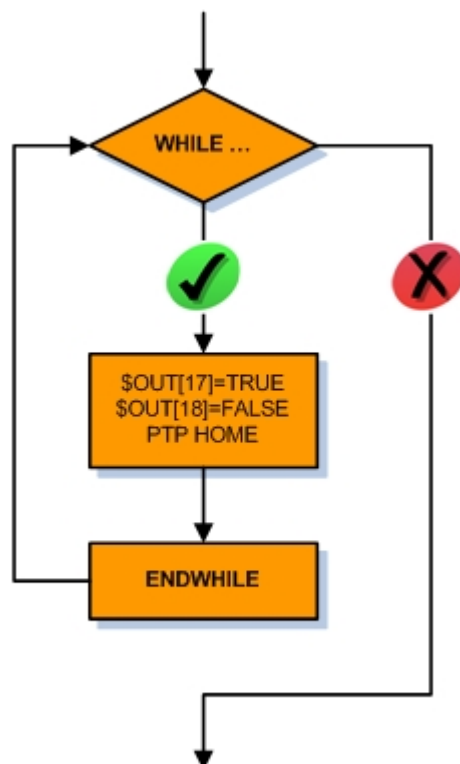


Рис. 13-4: План выполнения программы WHILE

Пример цикла WHILE: выход 17 принимает значение TRUE, выход 18 переключается на значение FALSE, а робот перемещается в положение Home только в том случае, если к началу цикла было выполнено условие (вход 22 = TRUE).

```

WHILE $IN[22]==TRUE
  $OUT[17]=TRUE
  $OUT[18]=FALSE
  PTP HOME
ENDWHILE

```

Цикл WHILE также можно запрограммировать в качестве цикла с подсчетом. Однако автоматический подсчет не выполняется (как в случае с FOR): изменение значения числовой переменной следует запрограммировать вручную.

### Неотклоняемый цикл

Цикл REPEAT – неотклоняемый цикл или цикл с последующей проверкой, при котором условие прерывания проверяется только после того, как была выполнена часть цикла с командами.

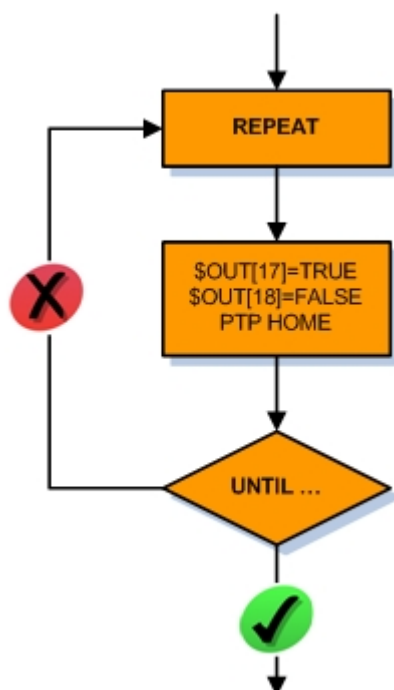


Рис. 13-5: План выполнения программы REPEAT

Пример цикла REPEAT: выход 17 принимает значение TRUE, выход 18 переключается на значение FALSE, а робот перемещается в положение Home. Лишь после этого проверяется условие.

```

REPEAT
  $OUT[17]=TRUE
  $OUT[18]=FALSE
  PTP HOME
UNTIL $IN[22]==TRUE

```

Цикл REPEAT также можно запрограммировать в качестве цикла с подсчетом. Однако автоматический подсчет не выполняется (как в случае с FOR): изменение значения числовой переменной следует запрограммировать вручную.



## 13.4 Обусловленные команды и различение ситуаций

Продолжи-  
тельность: 00:30:00  
Оснащение: Офисный ПК  
Информация:  
Сертификат:

### Условный переход

*Условный переход* (запрос IF) состоит из условия и двух разделов команд. При выполнении условия обрабатывается первая команда. Если условие не выполнено, выполняется вторая команда.

Однако запрос IF предлагает альтернативы:

- Второй раздел команд может быть пропущен: запрос IF без ELSE. Вследствие этого при невыполнении условия непосредственно после перехода программа продолжает работу.
- Несколько запросов IF могут быть вложены друг в друга (*многократный переход*): запросы обрабатываются по очереди, пока не будет выполнено какое-либо условие.

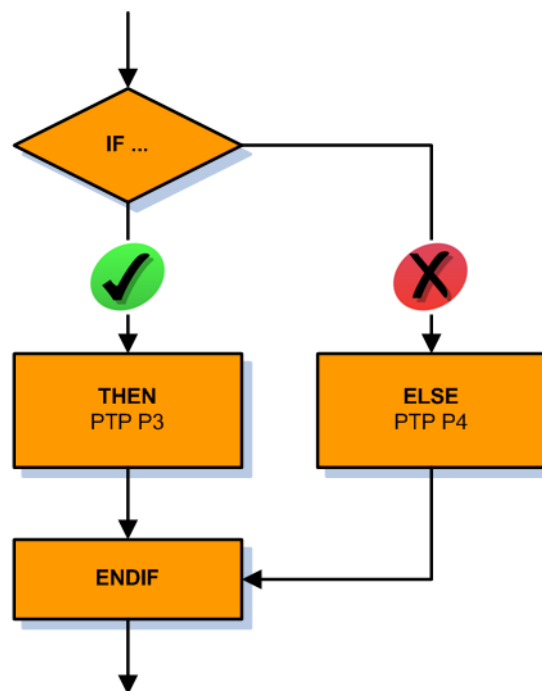


Рис. 13-6: План выполнения программы: переход IF

Пример запроса IF: если условие выполнено (вход 30 должен иметь значение TRUE), то робот перемещается в точку P3, в противном случае – в точку P4.

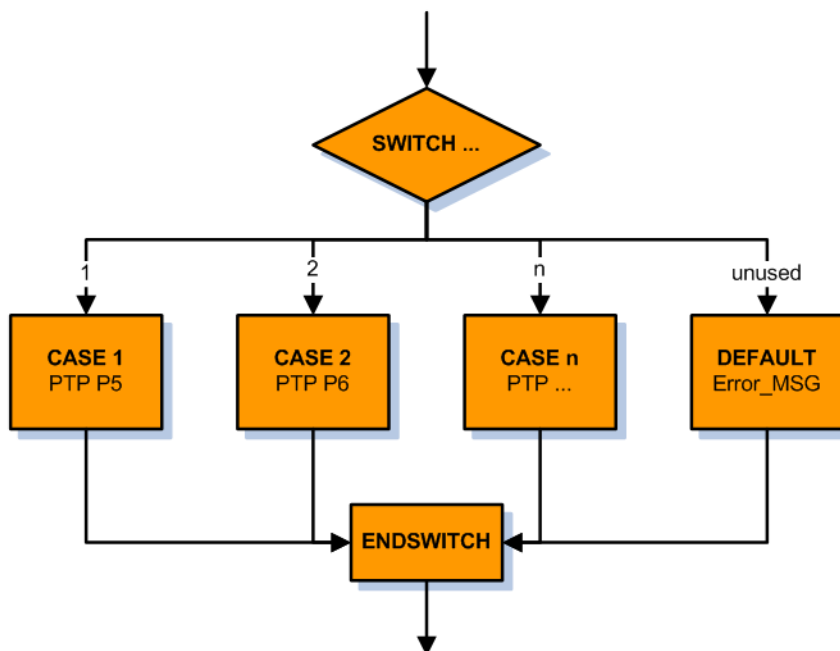
```

...
IF $IN[30]==TRUE THEN
  PTP P3
ELSE
  PTP P4
ENDIF
  
```

### Распределитель

Переход SWITCH является *распределителем* или *многократным переходом*. Сначала здесь оценивается выражение. Затем значение это-

го выражения сравнивается с значением одного из разделов случая (CASE). При совпадении выполняются указания соответствующего случая.



**Рис. 13-7: План выполнения программы: распределитель SWITCH – CASE**

Сначала целочисленная переменная (целое) с именем «status» проверяется на значимость. Если значение переменной равно «1», выполняется случай 1 (CASE 1): робот перемещается в точку P5. Если значение переменной равно «2», выполняется случай 2 (CASE 2): робот перемещается в точку P6. Если значение переменной не указано ни в одном из случаев (в данном случае все значения, кроме «1» и «2»), то выполняется ветвь DEFAULT: сообщение об ошибке.

```

INT status
...
SWITCH status
  CASE 1
    PTP P5
  CASE 2
    PTP P6
  ...
  DEFAULT
    ERROR_MSG()
ENDSWITCH
  
```

## 14 Подпрограммы и функции

### 14.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Локальные подпрограммы
- Глобальные подпрограммы
- Передача параметров в подпрограммы



Рис. 14-1: Смена главы

### 14.2 Работа с локальными подпрограммами

#### Определение локальных подпрограмм

- Локальные подпрограммы находятся после главной программы и обозначены DEF Имя\_подпрограммы( ) и END

```
DEF MY_PROG ( )
; это главная программа
...
END

DEF LOCAL_PROG1 ( )
; это локальная подпрограмма 1
...
END

DEF LOCAL_PROG2 ( )
; это локальная подпрограмма 2
...
END

DEF LOCAL_PROG3 ( )
; это локальная подпрограмма 3
...
END
```

- Файл SRC может состоять из максимум 255 локальных подпрограмм
- Локальные подпрограммы можно вызывать многократно
- Имя локальной программы требует скобок

#### Взаимосвязи при работе с локальными подпрограммами

- После обработки локальной подпрограммы выполняется возврат к следующей команде после вызова подпрограммы.

```

DEF MY_PROG ( )
; это главная программа
...
LOCAL_PROG1 ( )
...
END

-----

DEF LOCAL_PROG1 ( )
...
LOCAL_PROG2 ( )
...
END

-----

DEF LOCAL_PROG2 ( )
...
END

```

- Разрешается вкладывать друг в друга не более 20 подпрограмм.
- Координаты точек сохраняются в соответствующем списке DAT и доступны для всего файла.

```

DEF MY_PROG ( )
; это главная программа
...
PTP P1 Vel=100% PDAT1
...
END

-----

DEF LOCAL_PROG1 ( )
...
; это та же позиция, что и в главной программе
PTP P1 Vel=100% PDAT1
...
END

```

```

DEFDAT MY_PROG ( )
...
DECL E6POS XP1={X 100, Z 200, Z 300 ... E6 0.0}
...
ENDDAT

```

- С помощью RETURN можно завершить подпрограмму и выполнить возврат в вызывающий программный модуль.

```

DEF MY_PROG ( )
; это главная программа
...
LOCAL_PROG1 ( )
...
END

-----

DEF LOCAL_PROG1 ( )
...
IF $IN[12]==FALSE THEN
RETURN ; возврат в главную программу
ENDIF
...
END

```

#### Порядок действий при создании локальных подпрограмм

1. Выберите группу пользователей «Эксперты»
2. Отобразите строку DEF
3. Откройте файл SCR в редакторе

```

DEF MY_PROG ( )
...
END

```

4. С помощью курсора перейдите **под** строку END
5. Задайте новую локальную программную шапку с DEF, именем программы и скобками

```
DEF MY_PROG ( )
...
END
DEF PICK_PART ( )
```

6. Закройте новую подпрограмму командой END

```
DEF MY_PROG ( )
...
END
DEF PICK_PART ( )
END
```

7. После запуска Return устанавливается поперечная полоса между главной и подпрограммой

```
DEF MY_PROG ( )
...
END

DEF PICK_PART ( )
END
```

8. Теперь можно продолжить редактирование главной и подпрограммы
9. Закройте и сохраните программу

### 14.3 Работа с глобальными подпрограммами

#### Определение глобальных подпрограмм

- Глобальные подпрограммы имеют самостоятельные файлы SRC и DAT

```
DEF GLOBAL1 ( )
...
END
```

```
DEF GLOBAL2 ( )
...
END
```

- Глобальные подпрограммы можно вызывать многократно

#### Взаимосвязи при работе с локальными подпрограммами

- После обработки локальной подпрограммы выполняется возврат к следующей команде после вызова подпрограммы.

```
DEF GLOBAL1 ( )
...
GLOBAL2 ( )
...
END
```

```
DEF GLOBAL2 ( )
...
GLOBAL3 ( )
...
END
```

```
DEF GLOBAL3 ( )
...
END
```

- Разрешается вкладывать друг в друга не более 20 подпрограмм.
- Координаты точек сохраняются в соответствующем списке DAT и доступны только для соответствующей программы

```
DEF GLOBAL1 ( )
...
PTP P1 Vel=100% PDAT1
END
```

```
DEFDAT GLOBAL1 ( )
DECL E6POS XP1={X 100, Z 200, Z 300 ... E6 0.0}
ENDDAT
```

#### Различные координаты для P1 в Global2 ( )

```
DEF GLOBAL2 ( )
...
PTP P1 Vel=100% PDAT1
END
```

```
DEFDAT GLOBAL2 ( )
DECL E6POS XP1={X 800, Z 775, Z 999 ... E6 0.0}
ENDDAT
```

- С помощью RETURN можно завершить подпрограмму и выполнить возврат в вызывающий программный модуль

```
DEF GLOBAL1 ( )
...
GLOBAL2 ( )
...
END
```

```
DEF GLOBAL2 ( )
...
IF $IN[12]==FALSE THEN
RETURN ; Ruecksprung ins GLOBAL1 ( )
ENDIF
...
END
```

#### Порядок действий при создании глобальных подпрограмм

1. Выберите группу пользователей «Эксперты»
2. Создайте новую программу

```
DEF MY_PROG ( )
...
END
```

## 3. Повторно создайте вторую программу

```
DEF PICK_PART( )
...
END
```

## 4. Откройте файл SCR программы MY\_PROG в редакторе

## 5. Запрограммируйте вызов подпрограммы, используя имя программы и скобки

```
DEF MY_PROG( )
...
PICK_PART( )
...
END
```

## 6. Закройте и сохраните программу

## 14.4 Передача параметров в подпрограммы

### Передача параметров

Передача параметров в подпрограмму осуществляется при вызове программы в скобках:

```
DEF MY_PROG( )
...
CALC (K, L)
...
END
```

Для передачи параметров имеются две возможности:

Параметр	Описание
Параметр IN	Значение переменной остается неизменным в главной программе. Данный метод передачи также называется «Call by Value».
Параметр OUT	Подпрограмма считывает значение, изменяет его и записывает новое значение обратно в главную программу. Данный метод передачи также называется «Call by Reference».

```
DEF CALC(R:IN, S:OUT)
...
END
```



При передаче нескольких параметров следить за правильной последовательностью!

### Пример функции

Функция вызывается так же, как и подпрограмма, но не может стоять отдельно. Ее значение всегда должно быть соотнесено с переменной того же типа данных.

```
DEF MY_PROG( )
DECL REAL result, value
...
result = CALC(value)
...
END
```



Функция – это подпрограмма, которая посредством команды RETURN возвращает главной программе определенное значение и обладает определенным типом данных.

```

DEFFCT REAL calc(num:IN) ;типом данных функции является REAL
DECL REAL return_value, num
...
RETURN(return_value) ;передает значение обратно в главную программу
ENDFCT
    
```

**Стандартные функции KUKA**

Математические функции:

Описание	Функция KRL
Значение	ABS(x)
Корень	SQRT(x)
Синус	SIN(x)
Косинус	COS(x)
Тангенс	TAN(x)
Арккосинус	ACOS(x)
Арктангенс	ATAN2(y,x)

## 15 Переменные и описания

### 15.1 Обзор

В этом учебном разделе рассматриваются следующие элементы:

- хранение данных в KRL;
- работа с простыми типами данных;
- поля;
- структуры;
- перечислимый тип данных.



Рис. 15-1: Смена главы

### 15.2 Хранение данных в KRL

#### Работа с переменными в KRL

##### Общая информация о переменных

- В программировании роботов с помощью KRL переменная в самом общем смысле представляет собой просто контейнер для операндов (значений), которые возникают в процессе работы робота.
- Переменная имеет определенный присвоенный адрес в памяти компьютера.
- Переменная обозначается именем, не являющимся ключевым словом KUKA.
- Каждая переменная связана с определенным типом данных
- Перед использованием требуется объявление типа данных.
- В KRL различают локальные и глобальные переменные.

##### Время жизни переменных в KRL

- Под временем жизни понимают временной промежуток, в течение которого за переменной зарезервирована ячейка памяти.
- Временные переменные освобождают свою ячейку памяти при выходе из программы или функции.
- Переменные из списка данных сохраняют текущее значение в своей ячейке памяти на длительное время.

##### Действительность переменных в KRL

- Локально объявленные переменные доступны и видны только в этой программе.
- Глобальные переменные сохранены в центральном (глобальном) списке данных.
- Глобальные переменные могут также создаваться в локальном списке данных и при объявлении снабжаться ключевым словом **global**.

### Типы данных с KRL

- Тип данных обозначает объединение объектов в множество
- Предварительно заданные стандартные типы данных
  - Целые числа (INT)
  - Числа с плавающей точкой (REAL)
  - Логические значения (BOOL)
  - Отдельный символ (CHAR)
- Самостоятельно заданные стандартные типы данных
  - Перечислимый тип данных (ENUM)
  - Поля/массивы
  - Структуры (STRUC)
- Предварительно заданные типы данных KUKA
  - Позиция робота (POS или AXIS)
  - Режим работы (\$MODE\_OP)

### Использование переменных с KRL

#### Свойства имен

- Имена в KRL должны иметь максимально 24 символа.
- Имена KRL должны содержать буквы (A-Z), цифры (0-9), а также специальные символы «\_» и «\$».
- Имена в KRL не должны начинаться с цифр.
- Имена в KRL не должны быть ключевыми словами.
- Использование заглавных и строчных букв не имеет значения.

#### Типы данных с KRL

- Предварительно заданные стандартные типы данных

Простые типы данных	Целое число	Число с плавающей точкой	Логические значения	Отдельный символ
Ключевое слово	INT	REAL	BOOL	CHAR
Диапазон значений	$-2^{31} \dots (2^{31}-1)$	$\pm 1.1 \cdot 10^{-38} \dots \pm 3.4 \cdot 10^{38}$	TRUE/ FALSE	Набор символов ASCII
Примеры	-199 или 56	-0.0000123 или 3.1415	TRUE или FALSE	«A» или «q» или «7»

- Поля/массивы

```
Voltage[10] = 12.75
Voltage[11] = 15.59
```

- Сохранение нескольких переменных одинакового типа данных с помощью индекса
- Инициализация или изменение значений осуществляется с помощью индекса
- Максимальный размер поля зависит от потребности в памяти типа данных

- перечислимый тип данных.

```
color = #red
```

- Все значения перечислимого типа данных задаются при создании с помощью имени (текст)
- Система задает также очередность

- Максимальное количество элементов зависит от ячейки памяти
- **Составной тип данных/структура**

```
Date = {day 14, month 12, year 1996}
```

- Составной тип данных из компонентов различных типов данных
- Компоненты могут состоять из простых типов данных, а также из структур
- Возможен доступ к отдельным компонентам

#### Время жизни/действительность

- **Переменные, созданные в файле SRC, называются временными переменными и**
  - не всегда могут отображаться
  - действительны только в объявленной части программы
  - освобождают свою ячейку памяти при достижении последней строки программы (строки END)
- **Переменные в локальном файле DAT**
  - всегда могут отображаться во время выполнения программы соответствующего файла SRC
  - доступны во всем файле SRC, то есть также в локальных подпрограммах
  - могут также создаваться как глобальные переменные
  - сохраняют текущее значение в файле DAT и при повторном вызове начинают с сохраненного значения
- **Переменные в системном файле \$CONFIG.DAT**
  - доступны во всех программах (глобально);
  - могут отображаться всегда, даже если программа не активна;
  - сохраняют текущее значение в \$CONFIG.DAT.

#### Двойное объявление переменных

- Двойное объявление всегда возникает при использовании одинаковых последовательностей символов (имен).
- **Не** является двойным объявлением, если одинаковое имя используется в разных файлах SRC или DAT.
- Двойные объявления в одном и том же файле SRC и DAT недопустимы и вызывают сообщение об ошибке.
- Двойные объявления в файле SRC или DAT и \$CONFIG.DAT допускаются:
  - При вызове цикла программы, в котором была объявлена переменная, изменяется только локальное значение, а не значение в \$CONFIG.DAT.
  - При вызове стороннего цикла программы выполняется обращение только к значению из \$CONFIG.DAT и его изменение.

#### Системные данные KUKA

- Типы системных данных KUKA
  - Перечислимый тип данных, например режим работы (mode\_op)
  - Структура, например, дата/время (date)
- Системная информация поступает от системных переменных KUKA. Они
  - считывают текущую системную информацию,
  - изменяют текущую конфигурацию системы,
  - предварительно заданы и начинаются с символа «\$»
    - \$DATE (текущее время и дата)

- \$POS\_ACT (текущая позиция робота)
- ...

### 15.3 Работа с простыми типами данных

Далее разъясняется порядок создания, инициализации и изменения переменных. При этом используются только простые типы данных

#### Простые типы данных с KRL

- Целые числа (INT)
- Числа с плавающей точкой (REAL)
- Логические значения (BOOL)
- Отдельный символ (CHAR)

#### 15.3.1 Объявление переменных

##### Создание переменных

##### Объявление переменных

- Объявление всегда должно выполняться перед использованием
- Каждой переменной должен быть присвоен тип данных
- При задании имени необходимо соблюдать свойства имен
- Ключевое слово для объявления – **DECL**
- Ключевое слово DECL можно опускать при четырех простых типах данных
- Присваивание значения осуществляется с помощью указателя предварительного выполнения
- Объявление переменных может выполняться различным образом, поскольку это определяет время жизни и действительность соответствующей переменной:
  - объявление в файле SRC,
  - объявление в локальном файле DAT,
  - объявление в \$CONFIG.DAT,
  - объявление в локальном файле DAT с ключевым словом «global».
- Создание постоянных
  - Постоянные создаются с помощью ключевого слов **CONST**
  - Создавать постоянные можно только в списках данных

##### Принцип объявления переменной

##### Структура программы в файле SRC

- Раздел объявлений служит для объявления переменных
- Раздел инициализации часть начинается с первого присваивания значения, как правило, со строки INI
- В разделе команд выполняется присваивание или изменение значений

```
DEF main( )
; раздел объявлений
...
; раздел инициализации
INI
...
; раздел команд
PTP HOME Vel=100% DEFAULT
...
END
```

### Изменение стандартного вида

- Отображение строки DEF возможно только для группы пользователей «Эксперт»
- Это необходимо для перехода в раздел объявлений модулей перед строкой INI
- Для просмотра строки DEF и END, важно также при передаче переменных в подпрограммы

### Планирование объявления переменной

- Задание времени жизни
  - Файл **SRC**: временная переменная «умирает» в конце программного цикла
  - **Файл DAT**: переменная сохраняется после завершения
- Задание действительности/доступности
  - Локально в файле **SRC**: доступна только в том программном цикле, в котором она была объявлена. Таким образом, переменная доступна только между локальной строкой DEF и END (главная программа **или** локальная подпрограмма).
  - Локально в файле **DAT**: действительна во всей программе, то есть также в локальных подпрограммах.
  - **\$CONFIG.DAT**: доступна глобально, то есть доступ для чтения и записи возможен во всех программных циклах
  - Локально в файле **DAT** как глобальная переменная: доступна глобально, доступ для чтения и записи возможен во всех программных циклах, если файл DAT снабжается ключевым словом PUBLIC и при объявлении дополнительно указывается ключевое слово GLOBAL.
- Задание типа данных
  - **BOOL**: классические события «Да»/«Нет»
  - **REAL**: результаты арифметических операций для предотвращения ошибок при округлении
  - **INT**: классические переменные-счетчики для циклов с подсчетом или счетчиков изделий
  - **CHAR**: только один символ  
Строка или текст могут быть реализованы только как поле CHAR
- Задание имени и объявление
  - Использовать DECL, чтобы упростить читабельность программы
  - Использовать осмысленные, «говорящие» имена переменных
  - Не использовать криптографические имена или сокращения
  - Выбирать разумную длину имени, не использовать каждый раз 24 символа.

**Порядок действий при объявлении переменной с простым типом данных**

### Создание переменной в файле SRC

1. Выбрать группу пользователей «Эксперт»
2. Отобразить строку DEF
3. Открыть файл SRC в редакторе
4. Выполнить объявление переменной

```
DEF MY_PROG ( )
DECL INT counter
DECL REAL price
DECL BOOL error
DECL CHAR symbol
INI
...
END
```

5. Закрывать и сохранять программу

#### Создание переменной в файле DAT

1. Выбрать группу пользователей «Эксперт»
2. Открыть файл DAT в редакторе
3. Выполнить объявление переменной

```
DEFDAT MY_PROG
EXTERNAL DECLARATIONS
DECL INT counter
DECL REAL price
DECL BOOL error
DECL CHAR symbol
...
ENDDAT
```

4. Закрывать и сохранять список данных

#### Создание переменной в \$CONFIG.DAT

1. Выбрать группу пользователей «Эксперт»
2. В редакторе в папке SYSTEM открыть файл \$CONFIG.DAT

```
DEFDAT $CONFIG
BASISTECH GLOBALS
AUTOEXT GLOBALS
USER GLOBALS
ENDDAT
```

3. Выбрать формуляр «USER GLOBALS» и открыть его клавишей «Формуляр откр./закр.»
4. Выполнить объявление переменной

```
DEFDAT $CONFIG
...
;=====
; Userdefined Types
;=====
; Userdefined Externals
;=====
; Userdefined Variables
;=====
DECL INT counter
DECL REAL price
DECL BOOL error
DECL CHAR symbol
...
ENDDAT
```

5. Закрывать и сохранять список данных

#### Создание глобальной переменной в файле DAT

1. Выбрать группу пользователей «Эксперт»
2. Открыть файл DAT в редакторе
3. Добавить в список данных в шапке программы ключевое слово PUBLIC

```
DEFDAT MY_PROG PUBLIC
```

4. Выполнить объявление переменной



```

DEFDAT MY_PROG PUBLIC
EXTERNAL DECLARATIONS
DECL GLOBAL INT counter
DECL GLOBAL REAL price
DECL GLOBAL BOOL error
DECL GLOBAL CHAR symbol
...
ENDDAT

```

5. Закрывать и сохранять список данных

### 15.3.2 Инициализация переменных с простыми типами данных

#### Описание инициализации с помощью KRL

- После каждого объявления за переменной резервируется только одна ячейка памяти, значение всегда является недействительным.
- В файле SRC объявление и инициализация всегда выполняются на двух отдельных строках.
- В файле DAT объявление и инициализация всегда выполняются на одной строке.  
Постоянная всегда должна инициализироваться сразу при объявлении.
- Раздел инициализации начинается с первого присваивания значения.

#### Принцип инициализации

##### Инициализация целых чисел

- Инициализация десятичного числа

```
value = 58
```

- Инициализация двоичного числа

```
value = 'B111010'
```

Двоичный	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
Дес.	32	16	8	4	2	1

Расчет:  $1 \cdot 32 + 1 \cdot 16 + 1 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 0 \cdot 1 = 58$

- Инициализация шестнадцатеричного числа

```
value = 'H3A'
```

Шест.	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Дес.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Расчет:  $3 \cdot 16 + 10 = 58$

#### Порядок действий при инициализации с помощью KRL

##### Объявление и инициализация в файле SRC

1. Открытие файла SRC в редакторе
2. Объявление выполнено
3. Выполнение инициализации

```
DEF MY_PROG ( )
DECL INT counter
DECL REAL price
DECL BOOL error
DECL CHAR symbol
INI
counter = 10
price = 0.0
error = FALSE
symbol = "X"
...
END
```

#### 4. Закрытие и сохранение программы

##### Объявление и инициализация в файле DAT

1. Открытие файла DAT в редакторе
2. Объявление выполнено
3. Выполнение инициализации

```
DEFDAT MY_PROG
EXTERNAL DECLARATIONS
DECL INT counter = 10
DECL REAL price = 0.0
DECL BOOL error = FALSE
DECL CHAR symbol = "X"
...
ENDDAT
```

#### 4. Закрытие и сохранение списка данных

##### Объявление в файле DAT и инициализация в файле SRC

1. Открытие файла DAT в редакторе
2. Выполнение объявления

```
DEFDAT MY_PROG
EXTERNAL DECLARATIONS
DECL INT counter
DECL REAL price
DECL BOOL error
DECL CHAR symbol
...
ENDDAT
```

3. Закрытие и сохранение списка данных
4. Открытие файла SRC в редакторе
5. Выполнение инициализации

```
DEF MY_PROG ( )
...
INI
counter = 10
price = 0.0
error = FALSE
symbol = "X"
...
END
```

#### 6. Закрытие и сохранение программы

### 15.3.3 Манипуляция значениями переменных простых типов данных с помощью KRL

**Перечень возможностей для изменения значений переменных с помощью KRL**

Значения переменных изменяются в циклах программы (файл SRC) различным образом в зависимости от задачи. Далее приводятся самые распространенные методы. Манипуляции с помощью побитовых операций и стандартных функций возможны, но здесь подробно не разъясняются.

**Манипуляция с данными следующими способами:**

- **Основные арифметические действия**
  - (+) сложение
  - (-) вычитание
  - (\*) умножение
  - (/) деление
- **Операции сравнения**
  - (==) идентичный/равный
  - (<>) неравный
  - (>) больше
  - (<) меньше
  - (>=) больше или равно
  - (<=) меньше или равно
- **Логические операции**
  - (NOT) инвертирование
  - (AND) логическое И
  - (OR) логическое ИЛИ
  - (EXOR) исключающее ИЛИ
- **Побитовые операции**
  - (B\_NOT) побитное инвертирование
  - (B\_AND) побитная связь И
  - (B\_OR) побитная связь ИЛИ
  - (B\_EXOR) побитная связь – исключающее ИЛИ

**Стандартные функции**

- Абсолютные функции
- Функция корня
- Функция синуса и косинуса
- Функция тангенса
- Функция арккосинуса
- Функция арктангенса
- Несколько функций для манипуляции строками

**Взаимосвязи при манипуляции данными**

Изменение значения с использованием типа данных REAL и INT

- Округление в большую/меньшую сторону

```

; объявление
DECL INT A,B,C
DECL REAL R,S,T
; инициализация
A = 3           ; A=3
B = 5.5        ; B=6 (начиная с х.5 выполняется округление в большую
сторону)
C = 2.25       ; C=2 (округление в меньшую сторону)
R = 4          ; R=4.0
S = 6.5        ; S=6.5
T = C          ; T=2.0 (берется значение, округленное в меньшую
сторону)

```

■ Результаты арифметических операций (+;-;\*)

Операнды	INT	REAL
INT	INT	REAL
REAL	REAL	REAL

```

; объявление
DECL INT D,E
DECL REAL U,V
; инициализация
D = 2
E = 5
U = 0.5
V = 10.6
; раздел команд (манипуляция данными)
D = D*E ; D = 2 * 5 = 10
E = E+V ; E= 5 + 10.6 = 15.6 -> Округление в большую сторону E=16
U = U*V ; U= 0.5 * 10.6 = 5.3
V = E+V ; V= 16 + 10.6 = 26.6
    
```

■ Результаты арифметических операций (/)

Особенности при арифметических операциях с целочисленными значениями:

- При промежуточных результатах чистых операций с целыми числами все разряды после запятой отбрасываются.
- При присваивании значения целочисленной переменной результат округляется по обычным арифметическим правилам.

```

; объявление
DECL INT F
DECL REAL W
; инициализация
F = 10
W = 10.0
; раздел команд (манипуляция данными)
; INT / INT -> INT
F = F/2 ; F=5
F = 10/4 ; F=2 (10/4 = 2.5 -> Отброс разряда после запятой)
; REAL / INT -> REAL
F = W/4 ; F=3 (10.0/4=2.5 -> Округление в большую сторону)
W = W/4 ; W=2.5
    
```

**Операции сравнения**

Операторы сравнения позволяют создавать логические выражения. Результат сравнения всегда имеет тип данных Bool.

Оператор/ KRL	Описание	Допустимые типы данных
==	идентичный/ равный	INT, REAL, CHAR, BOOL
<>	неравный	INT, REAL, CHAR, BOOL
>	больше	INT, REAL, CHAR
<	меньше	INT, REAL, CHAR
>=	больше или равно	INT, REAL, CHAR
<=	меньше или равно	INT, REAL, CHAR

```

; объявление
DECL BOOL G, H
; раздел инициализации/команд
G = 10 > 10.1 ; G=FALSE
H = 10 / 3 == 3 ; H=TRUE
G = G <> H ; G=TRUE

```

### Логические операции

Логические операции позволяют создавать логические выражения. Результат такой операции всегда имеет тип данных Bool.

Операции		NOT A	A AND B	A OR B	A EXOR B
A=TRUE	B=TRUE	FALSE	TRUE	TRUE	FALSE
A=TRUE	B=FALSE	FALSE	FALSE	TRUE	TRUE
A=FALSE	B=TRUE	TRUE	FALSE	TRUE	TRUE
A=FALSE	B=FALSE	TRUE	FALSE	FALSE	FALSE

```

; объявление
DECL BOOL K, L, M
; раздел инициализации/команд
K = TRUE
L = NOT K ; L=FALSE
M = (K AND L) OR (K EXOR L) ; M=TRUE
L = NOT (NOT K) ; L=TRUE

```

Операторы приводятся в порядке их приоритета

Приоритет	Оператор
1	NOT (B_NOT)
2	Умножение (*); деление (/)
3	Сложение (+), вычитание (-)
4	AND (B_AND)
5	EXOR (B_EXOR)
6	OR (B_OR)
7	Любое сравнение (==; <>; ...)

```

; объявление
DECL BOOL X, Y
DECL INT Z
; раздел инициализации/команд
X = TRUE
Z = 4
Y = (4 * Z + 16 <> 32) AND X ; Y=FALSE

```

### Порядок действий при манипуляции данными

1. Установите тип данных для переменной или переменных
2. Определите действительность и время жизни переменной
3. Выполните объявление переменной
4. Инициализируйте переменную
5. Выполняйте манипуляции с переменной в циклах программы, то есть всегда в файле SCR

## 6. Закройте и сохраните файл SRC

**Команда останова**

Команда HALT останавливает программу. Последняя текущая команда перемещения еще полностью выполняется.

Программу можно продолжить кнопкой запуска. Затем выполняется следующая команда после HALT.



В программе прерывания программа останавливается только после полной обработки предварительного выполнения.

**15.4 Массивы/поля с KRL****Описание полей с KRL**

Поля, так же называемые массивами, могут использоваться для хранения нескольких переменных одного типа данных, различаемых по индексу

- Объем памяти полей конечен, то есть максимальный размер поля зависит от потребности в памяти соответствующего типа данных
- При объявлении должны быть известны размер поля и тип данных
- Начальный индекс KRL всегда начинается с 1
- Инициализация всегда может выполняться по отдельности
- Инициализация в файле SRC может выполняться также с помощью цикла

**Размеры полей**

- 1-мерное поле

```
dimension[4]= TRUE
```

- 2-мерное поле

```
dimension[2,1]= 3.25
```

- 3-мерное поле

```
dimension[3,4,1]= 21
```

- 4-мерное поле или выше в KRL **не** поддерживается

**Взаимосвязи при использовании полей**

Время жизни и действительность переменных полей такие же, как при использовании переменных простого типа данных

**Объявление поля**

- Создание в файле SRC

```
DEF MY_PROG ( )
DECL BOOL error[10]
DECL REAL value[50,2]
DECL INT parts[10,10,10]
INI
...
END
```

- Создание в списке данных (также \$CONFIG.DAT)

```
DEFDAT MY_PROG
EXTERNAL DECLARATIONS
DECL BOOL error[10]
DECL REAL value[50,2]
DECL INT parts[10,10,10]
...
ENDDAT
```

## Объявление и инициализация поля в файле SRC

- Каждое поле отдельно через вызов индекса

```
DECL BOOL error[10]
INI
error[1]=FALSE
error[2]=FALSE
error[3]=FALSE
error[3]=FALSE
error[4]=FALSE
error[5]=FALSE
error[6]=FALSE
error[7]=FALSE
error[8]=FALSE
error[9]=FALSE
error[10]=FALSE
```

- С помощью подходящих циклов

```
DECL BOOL error[10]
INI
DECL INT x
FOR x = 1 TO 10
  error[x]=FALSE
ENDFOR
```



После завершения цикла x имеет значение 11

## Инициализация поля в списке данных

- Каждое поле отдельно через вызов индекса и последующая индикация значения в списке данных

```
DEFDAT MY_PROG
EXTERNAL DECLARATIONS
DECL BOOL error[10]
error[1]=FALSE
error[2]=FALSE
error[3]=FALSE
error[4]=FALSE
error[5]=FALSE
error[6]=FALSE
error[7]=FALSE
error[8]=FALSE
error[9]=FALSE
error[10]=FALSE
```

- Недопустимое объявление и инициализация в списке данных

```
DEFDAT MY_PROG
EXTERNAL DECLARATIONS
DECL BOOL error[10]
DECL INT size = 32
error[1]=FALSE
error[2]=FALSE
error[3]=FALSE
error[4]=FALSE
error[5]=FALSE
error[6]=FALSE
error[7]=FALSE
error[8]=FALSE
error[9]=FALSE
error[10]=FALSE
```



Генерирует десять сообщений об ошибке «Запись начального значения отсутствует в разделе инициализации»





Все поле можно отобразить, если в главном меню выбрать **Индикация > Переменная > Отдельно** и в поле **Имя** указать имя переменных с квадратными скобками без индекса (напр., error[])

### Объявление полей в списке данных и инициализация в файле SRC

- Если поле создается в списке данных таким образом, текущие значения **нельзя** просмотреть в списке данных; текущие значения можно проверить **только** с помощью индикации переменных.

```
DEFDAT MY_PROG
EXTERNAL DECLARATIONS
DECL BOOL error[10]
```

```
DEF MY_PROG ( )
INI
  error[1]=FALSE
  error[2]=FALSE
  error[3]=FALSE
  ...
  error[10]=FALSE
```

или

```
DEF MY_PROG ( )
INI
FOR x = 1 TO 10
  error[x]=FALSE
ENDFOR
```

### Инициализация с помощью циклов

- 1-мерное поле

```
DECL INT parts[15]
DECL INT x
FOR x = 1 TO 15
  parts[x]= 4
ENDFOR
```

- 2-мерное поле

```
DECL INT parts_table[10,5]
DECL INT x, y
FOR x = 1 TO 10
  FOR y = 1 TO 5
    parts_table[x, y]= 6
  ENDFOR
ENDFOR
```

- 3-мерное поле

```
DECL INT parts_palette[5,4,3]
DECL INT x, y, z
FOR x = 1 TO 5
  FOR y = 1 TO 4
    FOR z = 1 TO 3
      parts_palette[x, y, z]= 12
    ENDFOR
  ENDFOR
ENDFOR
```

**Порядок действий при использовании массивов**

1. Задайте типы данных для поля
2. Определите действительность и время поля
3. Выполните объявление поля
4. Инициализируйте элементы поля

5. Выполняйте манипуляции с полем в циклах программы, то есть всегда в файле SCR
6. Закройте и сохраните файл SRC

```

DEF MY_PROG ( )
DECL REAL palette_size[10]
DECL INT counter
INI
; инициализация
FOR counter = 1 TO 10
  palette_size[counter] = counter * 1.5
ENDFOR
...
; изменение значения по отдельности
palette_size[8] = 13
...
; сравнение значений
IF palette_size[3] > 4.2 THEN
...

```

## 15.5 Структуры с KRL

Переменные с несколькими отдельными данными

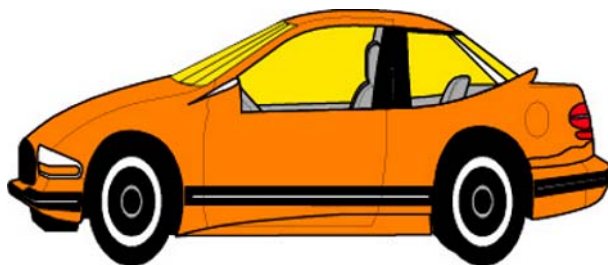


Рис. 15-2

Составной тип данных: **Структура**

- Поля позволяют объединять переменные одинакового типа данных. Но в реальности большинство переменных состоит из разных типов данных.
- Так, такие параметры автомобиля, как мощность двигателя или километраж, являются целочисленными (Integer). Цена скорее принадлежит к типу Real. Напротив, наличие кондиционера является скорее типом данных Bool.
- Все вместе описывает автомобиль.
- Структура может быть определена с помощью самого ключевого слова STRUC.
- Структура представляет собой совокупность разных типов данных.

```
STRUC CAR_TYPE INT motor, REAL price, BOOL air_condition
```

- Структура сначала должна быть определена, тогда она может использоваться в дальнейшем.

Использование структуры

**Доступность/определение структуры**

- Локально созданные структуры становятся недействительными при достижении строки END
- Структуры, используемые в нескольких программах, должны объявляться в \$CONFIG.DAT
- Самостоятельно заданные структуры для большей наглядности должны оканчиваться **TYPE**
- В структуре могут использоваться простые типы данных INT, REAL, BOOL и CHAR.

```
STRUC CAR_TYPE INT motor, REAL price, BOOL air_condition
```

- В структуру можно встраивать поля CHAR.

```
STRUC CAR_TYPE INT motor, REAL price, BOOL air_condition, CHAR  
car_model[15]
```

- В структуре могут также использоваться известные структуры, например позиция POS.

```
STRUC CAR_TYPE INT motor, REAL price, BOOL air_condition, POS  
car_pos
```

- После определения структуры должна быть объявлена соответствующая рабочая переменная.

```
STRUC CAR_TYPE INT motor, REAL price, BOOL air_condition  
DECL CAR_TYPE my_car
```

### Инициализация/изменение структуры

- Инициализация может выполняться с помощью скобок.
- При инициализации с помощью скобок разрешается использовать только постоянные (фиксированные значения).
- Очередность присваивания значений не играет роли.

```
my_car = {motor 50, price 14999.95, air_condition TRUE}
```

```
my_car = {price 14999.95, motor 50, air_condition TRUE}
```

Система выполняет сортировку в последовательности, в которой была создана структура.

Здесь: STRUC CAR\_TYPE INT motor, REAL price, BOOL air\_condition

- motor
- price
- air\_condition

- В структуре не обязательно должны быть указаны все ее элементы.
- Структура инициализируется с помощью элемента структуры.
- Для неинициализированных значений (будет) установлено «неизвестно».

```
my_car = {motor 75} ; Значение неизвестно для цены и кондиционера
```

- Инициализация может выполняться также с помощью точки-разделителя.

```
my_car.price = 9999.0
```

- Элемент структуры в отдельности может быть в любой момент изменен с помощью точки-разделителя. При этом другие элементы структуры не удаляются и не изменяются.

```
my_car = {price 14999.95, motor 50, air_condition TRUE}  
my_car.price = 12000.0 ; Значение для двигателя и кондиционера  
остаются неизменными
```

**Но:**

```
my_car = {price 14999.95, motor 50, air_condition TRUE}  
my_car = {price 12000.0} ; Значение для двигателя и кондиционера  
удаляется
```

KUKA часто в работе используется предварительно заданные структуры, сохраненные в системе. Примеры этого можно найти в позициях и про-

граммировании сообщений.

### Предварительно заданные структуры KUKA для работы с позициями

- **AXIS:** STRUC AXIS REAL A1, A2, A3, A4, A5, A6
- **E6AXIS:** STRUC E6AXIS REAL A1, A2, A3, A4, A5, A6, E1, E2, E3, E4, E5, E6
- **FRAME:** STRUC FRAME REAL X, Y, Z, A, B, C
- **POS:** STRUC POS REAL X, Y, Z, A, B, C INT S,T
- **E6POS:** STRUC E6POS REAL X, Y, Z, A, B, C, E1, E2, E3, E4, E5, E6 INT S,T

### Инициализация структуры с позицией

- При инициализации с помощью скобок разрешается использовать только постоянные (фиксированные значения)

```
STRUC CAR_TYPE INT motor, REAL price, BOOL air_condition, POS
car_pos
DECL CAR_TYPE my_car
my_car = {price 14999.95, motor 50, air_condition TRUE, car_pos {X
1000, Y 500, A 0}}
```

- Инициализация может выполняться также с помощью точки-разделителя

```
my_car.price = 14999.95
my_car.motor = 50
my_car.air_condition = TRUE
my_car.car_pos = {X 1000, Y 500, A 0}
```

- При инициализации с помощью точки-разделителя можно использовать также переменные

```
my_car.price = my_price*0.85
my_car.car_pos.X = x_value
my_car.car_pos.Y = 750+y_value
```

## Создание структуры

### 1. Определение структуры

```
STRUC CAR_TYPE INT motor, REAL price, BOOL air_condition
```

### 2. Объявление рабочей переменной

```
DECL CAR_TYPE my_car
```

### 3. Инициализация рабочей переменной

```
my_car = {motor 50, price 14999.95, air_condition TRUE}
```

### 4. Изменение значений и/или сравнение значений рабочих переменных

```
my_car.price = 5000.0
```

```
my_car.price = value_car
```

```
IF my_car.price >= 20000.0 THEN
...
ENDIF
```

## 15.6 Перечислимый тип данных ENUM

Текст как значение переменной

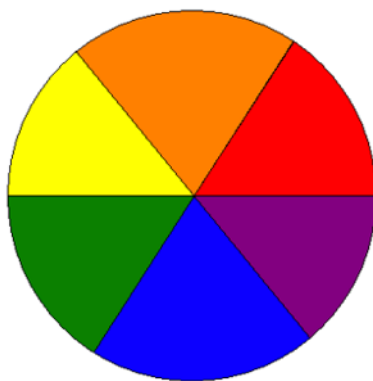


Рис. 15-3

- Перечислимый тип данных состоит из ограниченного количества постоянных, таких как зеленый, желтый или синий

```
ENUM COLOR_TYPE green, blue, red, yellow
```

- Постоянные представляют собой произвольно выбираемые имена
- Постоянные задаются программистом
- Перечислимый тип данных сначала должен быть определен, тогда он может использоваться в дальнейшем
- Рабочая переменная, например цвет ящика, типа COLOR\_TYPE всегда может принимать только одно значение постоянной
- Присваивание значения постоянной всегда выполняется с помощью символа #

Использование перечислимого типа данных

### Доступность/использование

- Можно использовать только известные постоянные
- Перечислимый тип данных можно дополнять с какой угодно частотой
- Перечислимый тип данных может использоваться самостоятельно

```
ENUM COLOR_TYPE green, blue, red, yellow
```

- Перечислимый тип данных можно интегрировать в структуру

```
ENUM COLOR_TYPE green, blue, red, yellow
STRUC CAR_TYPE INT motor, REAL price, COLOR_TYPE car_color
```

### Действительность/время жизни

- Локально созданные перечислимые типы данных становятся недействительными при достижении строки END
- Перечислимые типы данных, используемые в нескольких программах, должны объявляться в \$CONFIG.DAT

### Присваивание имени

- Имена перечислимых типов данных и их постоянные должны быть «говорящими»
- Запрещено использовать ключевые слова
- Самостоятельно заданные перечислимые типы данных для большей наглядности должны оканчиваться TYPE

Создание перечислимого типа данных

1. Определение перечислимых переменных и постоянных

```
ENUM LAND_TYPE de, be, cn, fr, es, br, us, ch
```

2. Объявление рабочей переменной

```
DECL LAND_TYPE my_land
```

### 3. Инициализация рабочей переменной

```
my_land = #be
```

### 4. Сравнение значений рабочей переменной

```
IF my_land == #es THEN  
...  
ENDIF
```



## 16 Программирование перемещений с помощью KRL

### 16.1 Обзор

В этом учебном разделе рассматриваются следующие элементы:

- программирование перемещений с помощью KRL;
- программирование относительных перемещений с помощью KRL;
- расчет и манипуляция позициями робота;
- целенаправленное изменение состояния и поворота.



Рис. 16-1: Смена главы

### 16.2 Программирование перемещений с помощью KRL

#### Определение перемещения

Необходимые данные для перемещения

- Вид перемещения – PTP, LIN, CIRC
- Целевая позиция и при необходимости вспомогательная позиция
- Точный останов или сглаживание
- При необходимости расстояние сглаживания
- Скорость и - PTP (%) и перемещение по траектории (м/с)
- Ускорение
- Инструмент - TCP и нагрузка
- Рабочая база
- Управляемый роботом или внешний инструмент
- Контроль ориентации при перемещениях по траектории
- При необходимости расстояние сглаживания
- Угол круга при круговом перемещении CIRC

#### Принцип программирования перемещений

Вид перемещения **PTP**

- PTP *Целевая точка* <C\_PTP <Сглаживание траектории>>



Элемент	Описание
Целевая точка	<p>Тип: POS, E6POS, AXIS, E6AXIS, FRAME</p> <p>Целевая точка может указываться в прямоугольной системе координат или по осям. Прямоугольные координаты действительны для системы координат BASE.</p> <p>Если указаны не все компоненты целевой точки, система управления применяет для отсутствующих компонентов значения предыдущей позиции.</p>
C_PTP	<p>Включает сглаживание целевой точки.</p> <p>При сглаживании PTP-PTP достаточно указания C_PTP.</p>
Сглаживани е траектории	<p>Только для сглаживания PTP-CP. Этот параметр определяет самый ранний момент, в который начинается сглаживание. Доступны варианты ввода:</p> <ul style="list-style-type: none"> <li>■ C_DIS            Параметр расстояния (по умолчанию): Сглаживание начинается самое раннее в тот момент, когда расстояние до целевой точки становится меньше значения \$APO.CDIS.</li> <li>■ C_ORI            Параметр ориентирования: Сглаживание начинается самое раннее в тот момент, когда доминирующий угол ориентации становится меньше значения \$APO.CORI.</li> <li>■ C_VEL            Параметр скорости: Сглаживание начинается самое раннее в тот момент, когда скорость в фазе торможения в направлении целевой точки становится меньше значения \$APO.CVEL.</li> </ul>

- Робот подводится к позиции из файла DAT; позиция предварительно заучена с помощью встроенного формуляра и сглаживает эту точку P3

```
PTP XP3 C_PTP
```

- Робот перемещается во введенную позицию
  - По осям (AXIS или E6AXIS)

```
PTP {A1 0, A2 -80, A3 75, A4 30, A5 30, A6 110}
```

- Позиция в пространстве (с активным инструментом и базой)

```
PTP {X 100, Y -50, Z 1500, A 0, B 0, C 90, S 3, T 35}
```

- Робот перемещается только при вводе одного или нескольких агрегатов

```
PTP {A1 30} ; Только A1 перемещается на 30°
```

```
PTP {X 200, A 30} ; Только по X на 200 мм и A на 30°
```

#### Вид перемещения LIN

- LIN *Целевая точка* <Сглаживание траектории>

Элемент	Описание
Целевая точка	<p>Тип: POS, E6POS, FRAME</p> <p>Если указаны не все компоненты целевой точки, система управления применяет для отсутствующих компонентов значения предыдущей позиции.</p> <p>Данные состояния (Status) и поворота (Turn) внутри одной целевой точки типа POS или E6POS при перемещениях LIN (так же, как и при перемещениях CIRC) игнорируются.</p> <p>Координаты действительны для системы координат BASE.</p>
Сглаживание траектории	<p>Этот параметр включает сглаживание целевой точки. Одновременно он определяет самый ранний момент, в который начинается сглаживание. Возможные варианты ввода:</p> <ul style="list-style-type: none"> <li>■ C_DIS Параметр расстояния: Сглаживание начинается самое раннее в тот момент, когда расстояние до целевой точки становится меньше значения \$APO.CDIS.</li> <li>■ C_ORI Параметр ориентирования: Сглаживание начинается самое раннее в тот момент, когда доминирующий угол ориентации становится меньше значения \$APO.CORI.</li> <li>■ C_VEL Параметр скорости: Сглаживание начинается самое раннее в тот момент, когда скорость в фазе торможения в направлении целевой точки становится меньше значения \$APO.CVVEL.</li> </ul>

- Робот перемещается в рассчитанную позицию и сглаживает эту точку ABLAGE[4]

```
LIN ABLAGE[4] C_DIS
```

#### Вид перемещения CIRC

- CIRC *Вспомогательная точка, целевая точка*<, CA *Угол круга*>  
<Сглаживание траектории>

Элемент	Описание
<i>Вспомогательная точка</i>	<p>Тип: POS, E6POS, FRAME</p> <p>Если указаны не все компоненты вспомогательной точки, система управления применяет для отсутствующих компонентов значения предыдущей позиции.</p> <p>Угол ориентации, а также данные состояния (Status) и поворота (Turn) внутри одной вспомогательной точки всегда игнорируются.</p> <p>Сглаживание вспомогательной точки невозможно. Перемещение всегда выполняется точно в нее.</p> <p>Координаты действительны для системы координат BASE.</p>
<i>Целевая точка</i>	<p>Тип: POS, E6POS, FRAME</p> <p>Если указаны не все компоненты целевой точки, система управления применяет для отсутствующих компонентов значения предыдущей позиции.</p> <p>Данные состояния (Status) и поворота (Turn) внутри одной целевой точки типа POS или E6POS при перемещениях CIRC (так же, как и при перемещениях LIN) игнорируются.</p> <p>Координаты действительны для системы координат BASE.</p>

Элемент	Описание
Угол круга	<p>Задаёт общий угол кругового перемещения. Это позволяет продлить движение за пределы запрограммированной целевой точки или же укоротить его. В данном случае действительная целевая точка больше не соответствует запрограммированной целевой точке.</p> <p>Блок: град. Без ограничения; в частности может быть запрограммирован угол круга более 360°.</p> <ul style="list-style-type: none"> <li>■ Угол круга положительный: круговая траектория проходит в направлении начальная точка &gt; вспомогательная точка &gt; целевая точка.</li> <li>■ Угол круга отрицательный: круговая траектория проходит в направлении начальная точка &gt; целевая точка &gt; вспомогательная точка.</li> </ul>
Сглаживание траектории	<p>Этот параметр включает сглаживание целевой точки. Одновременно он определяет самый ранний момент, в который начинается сглаживание. Возможные варианты ввода:</p> <ul style="list-style-type: none"> <li>■ C_DIS Параметр расстояния: Сглаживание начинается самое раннее в тот момент, когда расстояние до целевой точки становится меньше значения \$APO.CDIS.</li> <li>■ C_ORI Параметр ориентирования: Сглаживание начинается самое раннее в тот момент, когда доминирующий угол ориентации становится меньше значения \$APO.CORI.</li> <li>■ C_VEL Параметр скорости: Сглаживание начинается самое раннее в тот момент, когда скорость в фазе торможения в направлении целевой точки становится меньше значения \$APO.CVEL.</li> </ul>

- Робот подводится к позициям из файла DAT; позиции предварительно заучены с помощью встроенного формуляра и имеют угол круга 190°

CIRC XP3, XP4, CA 190

- Угол круга **CA**



В действительной целевой точке занимается ориентация, заученная в запрограммированной целевой точке.

- **Угол круга положительный (CA>0):** Круг проходит в запрограммированном направлении: начальная точка – вспомогательная точка – целевая точка

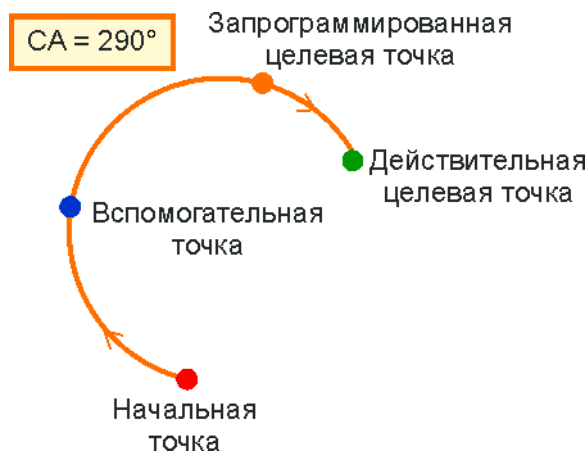


Рис. 16-2: Угол круга CA = +290°

- Угол круга отрицательный (CA<0): Круг проходится против запрограммированного направления: начальная точка – целевая точка – вспомогательная точка

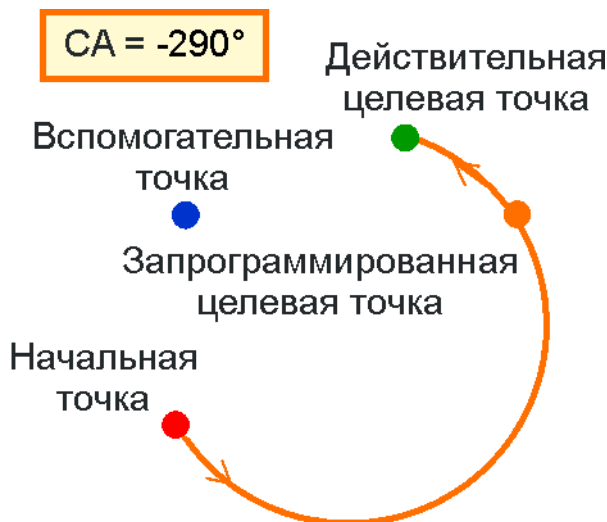


Рис. 16-3: Угол круга CA = -290°

**Функция параметров перемещения**

**Предварительные настройки для программирования перемещения**

- Можно применять следующие настройки:
  - из прохода строки INI
  - из последнего встроенного формуляра
  - из последних настроек соответствующих системных переменных
- Изменение или инициализация соответствующих системных переменных

**Системные переменные параметров перемещения**

- **Инструмент:** \$TOOL и \$LOAD
  - Активация измеренной TCP

```
$TOOL = tool_data[x] ; x = 1...16
```

- Активация соответствующих данных нагрузки

```
$LOAD = load_data[x] ; x = 1...16
```

- **Отсчетная база/рабочая база:** \$BASE
  - Активация измеренной базы

```
$BASE = base_data[x] ; x = 1...32
```

- **Управляемый роботом или внешний инструмент: \$IPO\_MODE**

- Управляемый роботом инструмент

```
$IPO_MODE = #BASE
```

- Внешний инструмент

```
$IPO_MODE = #TCP
```

- **Скорость:**

- При перемещении PTP

```
$VEL_AXIS[x] ; x=1...8 для каждой оси
```

- При перемещениях по траектории LIN или CIRC

```
$VEL.CP = 2.0 ; [m/s] Скорость перемещения по траектории
```

```
$VEL.ORI1 = 150 ; [°/s] Скорость поворота
```

```
$VEL.ORI2 = 200 ; [°/s] Скорость вращения
```



Направлением удара инструмента в большинстве случаев является ось X. Скорость вращения – это вращение вокруг оси X под углом C. При скорости поворота выполняется поворот на два других угла (A и B).

- **Ускорение**

- При перемещении PTP

```
$ACC_AXIS[x] ; x=1...8 для каждой оси
```

- При перемещениях по траектории LIN или CIRC

```
$ACC.CP = 2.0 ; [m/s²] ускорение перемещения по траектории
```

```
$ACC.ORI1 = 150 ; [°/s²] ускорение поворота
```

```
$ACC.ORI2 = 200 ; [°/s²] ускорение вращения
```

- **Расстояние сглаживания**

- Только при перемещении PTP: **C\_PTP**

```
$APO_CPTP = 50 ; величина сглаживания в [%] при C_PTP
PTP XP3 C_PTP
```

- При перемещениях по траектории LIN, CIRC и при PTP: **C\_DIS**

Расстояние до целевой точки должно быть меньше значения \$APO.CDIS

```
$APO.CDIS = 250.0 ; [mm] расстояние
PTP XP3 C_DIS
LIN XP4 C_DIS
```

- При перемещениях по траектории LIN, CIRC: **C\_ORI**

Доминирующий угол ориентации должен быть меньше значения \$APO.CORI

```
$APO.CORI = 50.0 ; [°] угол
LIN XP4 C_ORI
```

- При перемещениях по траектории LIN, CIRC: **C\_VEL**

Скорость в фазе торможения в направлении целевой точки должна быть меньше \$APO.CVEL

```
$APO.CVEL = 75.0 ; [%] процент
LIN XP4 C_VEL
```

- **Контроль ориентации: только для LIN и CIRC**

- При LIN и CIRC: **\$ORI\_TPYE**

```
$ORI_TYPE = #CONSTANT
```

Ориентация во время перемещения по траектории остается постоянной. Запрограммированная в конечной точке ориентация игнорируется.

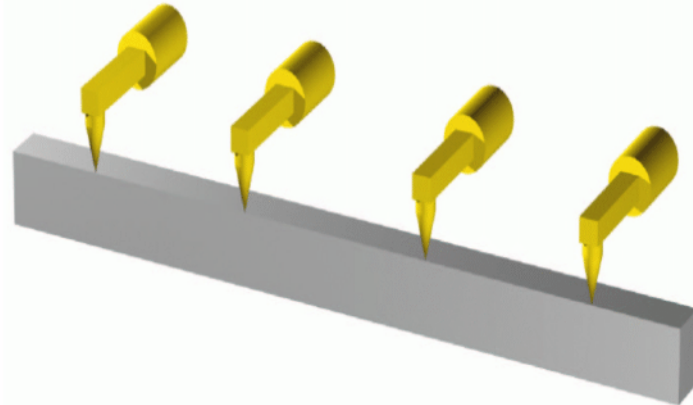


Рис. 16-4: Постоянный контроль ориентации

```
$ORI_TYPE = #VAR
```

Во время перемещения по траектории ориентация непрерывно изменяется до получения ориентации целевой точки.

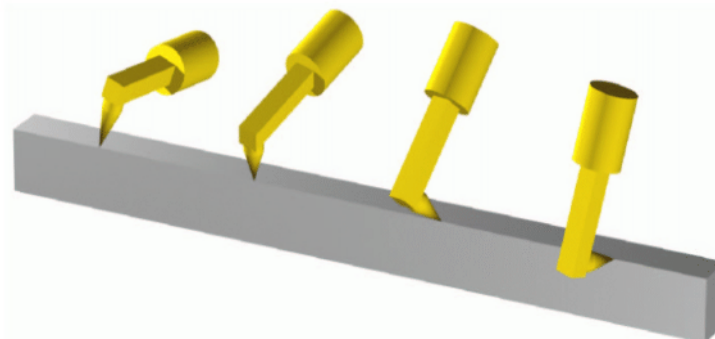


Рис. 16-5: Стандартный или Рука РТР

```
$ORI_TYPE = #JOINT
```

Во время перемещения по траектории ориентация инструмента непрерывно изменяется от начальной к конечной позиции. Это происходит из-за линейного перевода углов ручной оси. Эта опция позволяет избежать проблемы сингулярности руки, поскольку отсутствует контроль ориентации посредством вращения и поворота вокруг направления удара инструмента.

- Только при CIRC: **\$CIRC\_TPYE**



Переменная **\$CIRC\_TPYE** **не имеет значения**, если **\$ORI\_TYPE = #JOINT** выполняется линейный перевод углов ручной оси.

```
$CIRC_TYPE = #PATH
```

Контроль ориентации по отношению к траектории при круговом перемещении

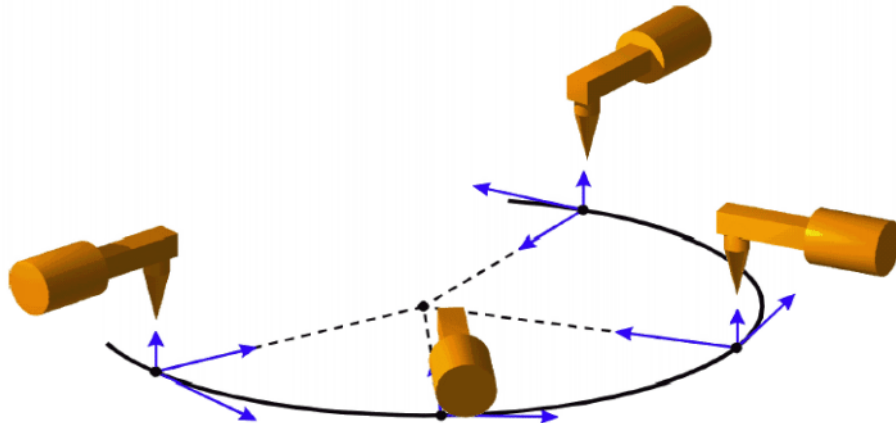


Рис. 16-6: Постоянная ориентация, относительно траектории

```
$CIRC_TYPE = #BASE
```

Пространственный контроль ориентации при круговом перемещении

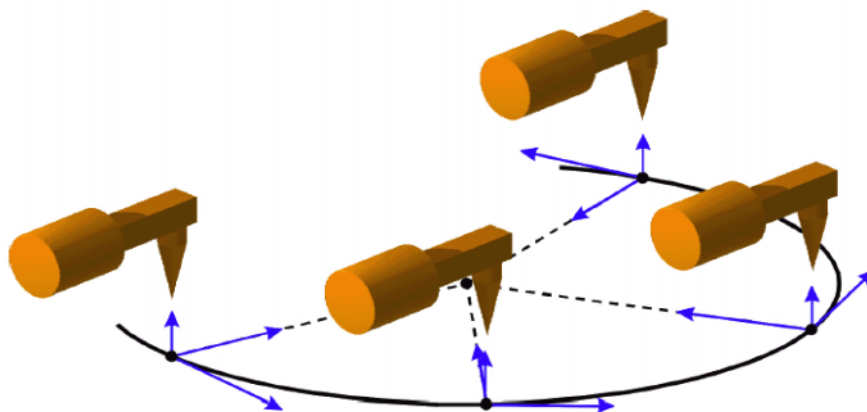


Рис. 16-7: Постоянная ориентация, относительно базы

**Порядок действий при программировании перемещений с помощью KRL**

1. Выбрав уровень эксперта, загрузите программу в редактор нажатием на кнопку **Открыть**
2. Проверьте предварительные настройки для программирования перемещения и примите их или выполните повторную инициализацию:
  - Инструмент (\$TOOL и \$LOAD)
  - Настройки базы (\$BASE)
  - Управляемый роботом или внешний инструмент (\$IPO\_MODE)
  - Скорость
  - Ускорение
  - При необходимости расстояние сглаживания
  - При необходимости контроль ориентации



3. Создайте команду перемещения из следующих компонентов:
  - Вид перемещения (PTP, LIN, CIRC)
  - Целевая точка (при CIRC также вспомогательная точка)
  - При CIRC угол круга (CA)
  - Активируйте сглаживание (C\_PTP, C\_DIS, C\_ORI, C\_VEL)
4. При повторном перемещении назад к пункту 3
5. Закройте редактор и сохраните

## 16.3 Расчет или манипуляция позициями робота

### Описание

Целевые позиции робота

- Сохраняются в структурах:
- Имеются указанные ниже предустановленные структуры.
  - **AXIS**  
Осовой угол A1 ... A6
  - **E6AXIS**  
Осовой угол A1 ... A6 и E1 ... E6
  - **POS**  
Позиция (X, Y, Z), ориентация (A, B, C) с состоянием и поворотом (S, T)
  - **E6POS**  
Позиция (X, Y, Z), ориентация (A, B, C) с состоянием и поворотом (S, T) и E1 ... E6
  - **FRAME -**  
Позиция (X, Y, Z), ориентация (A, B, C)
- Возможно манипулирование существующими позициями из файла DAT.
- Можно целенаправленно изменять отдельные агрегаты на существующих позициях с помощью точки-разделителя.

### Принцип

**УВЕДОМЛЕНИЕ** При расчете важно соблюдать правильные настройки TOOL и BASE, а затем активировать их при программировании перемещения. При несоблюдении возможно выполнение неожиданных перемещений и столкновение.

#### Важные системные переменные

- **\$POS\_ACT**: текущая позиция робота. Переменная (E6POS) описывает заданную позицию TCP относительно системы координат BASE.
- **\$AXIS\_ACT**: текущая позиция робота по осям (заданное значение). Переменная (E6AXIS) содержит текущие углы осей или позиции осей.

#### Расчет абсолютной целевой позиции

- Однократное изменение позиции из файла DAT.

```
XP1.x = 450 ; новое значение X 450 мм
XP1.z = 30.0*distance ; рассчитывается новое значение Z
PTP XP1
```

- Изменение позиции из файла DAT при каждом проходе

```
; значение X каждый раз смещается на 450 мм
XP2.x = XP2.x + 450
PTP XP2
```

- Позиция принимается и сохраняется в переменной

```

myposition = XP3
myposition.x = myposition.x + 100 ; к значению x прибавляется 100 мм
myposition.z = 10.0*distance ; расчет нового значения Z
myposition.t = 35 ; установка значения поворота
PTP XP3 ; позиция не была изменена
PTP myposition ; рассчитанная позиция

```

### Порядок действий

1. Выбрав уровень эксперта, загрузите программу в редактор нажатием на кнопку **Открыть**
2. Выполните расчет/манипуляцию позиции. Промежуточно сохраните вновь рассчитанные значения в новой переменной
3. Проверьте предварительные настройки для программирования перемещения и примите их или выполните повторную инициализацию:
  - Инструмент (\$TOOL и \$LOAD)
  - Настройки базы (\$BASE)
  - Управляемый роботом или внешний инструмент (\$IPO\_MODE)
  - Скорость
  - Ускорение
  - При необходимости расстояние сглаживания
  - При необходимости контроль ориентации
4. Создайте команду перемещения из следующих компонентов:
  - Вид перемещения (PTP, LIN, CIRC)
  - Целевая точка (при CIRC также вспомогательная точка)
  - При CIRC угол круга (CA)
  - Активируйте сглаживание (C\_PTP, C\_DIS, C\_ORI, C\_VEL)
5. При повторном перемещении назад к пункту 3
6. Закройте редактор и сохраните

## 16.4 Целенаправленное изменение битов состояния и поворота

### Описание

- Значения позиции (X, Y, Z) и ориентации (A, B, C) TCP являются недостаточными для того, чтобы однозначно задать позицию робота, поскольку при одинаковой точке TCP все же возможно несколько осевых положений. Состояние (Status) и поворот (Turn) служат для того, чтобы задать однозначное осевое положение из нескольких возможных осевых положений.

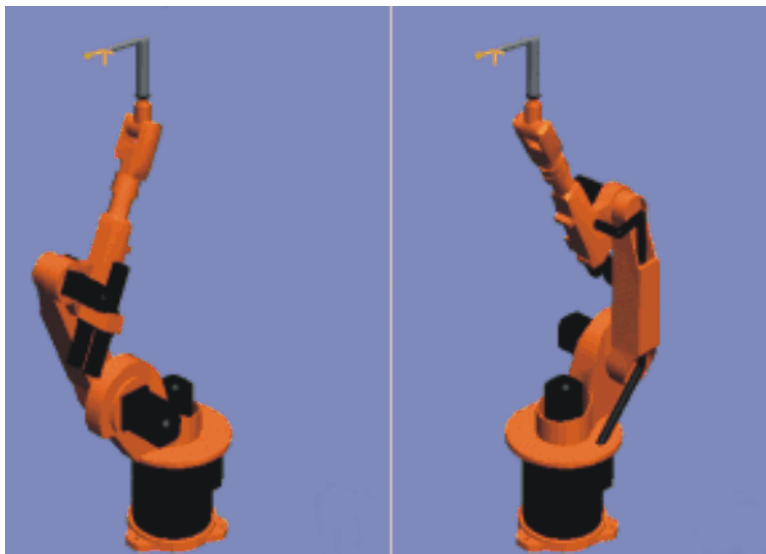


Рис. 16-8: Пример: TCP одинаковая, осевое положение разное

- Status (S) и Turn (T) являются компонентами типов данных POS и E6POS:

```
STRUC POS REAL X, Y, Z, A, B, C, INT S, T
```

```
STRUC E6POS REAL X, Y, Z, A, B, C, E1, E2, E3, E4, E5, E6, INT S, T
```

- Система управления роботом учитывает запрограммированные значения состояния и поворота **только при перемещениях PTP**. При перемещениях CP они игнорируются.
- Поэтому первая команда перемещения в программе KRL должна быть одной из следующих команд, чтобы для робота было определено однозначное исходное положение:
  - полная команда PTP типа POS или E6POS;
  - или полная команда PTP типа AXIS или E6AXIS.

«Полная» означает, что должны быть указаны все компоненты целевой точки. Позиция HOME по умолчанию сохранена в системе управления роботом.
- В следующих командах состояние (Status) и поворот (Turn) могут быть опущены:
  - Система управления роботом сохраняет предыдущее значение состояния.
  - Точка поворота при перемещениях CP определяется из траектории.
  - При перемещениях PTP система управления роботом выбирает значение поворота, которое обеспечивает максимально короткий путь (то есть не повреждает программный концевой выключатель и одновременно ближайшее к начальному углу).

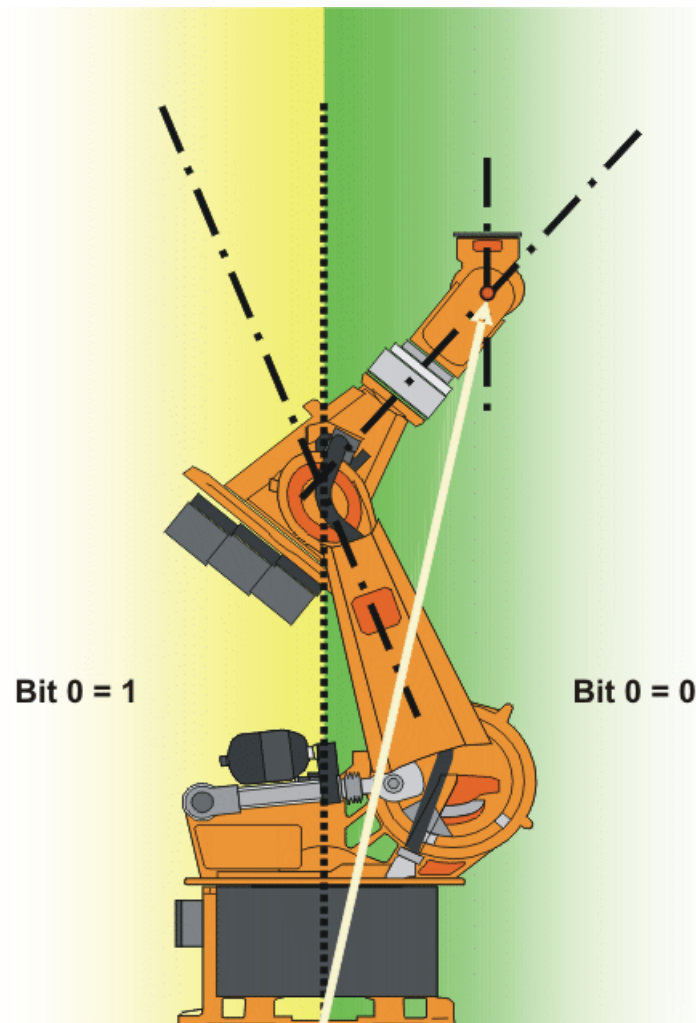
**Функция**

**СОСТОЯНИЕ**

- Указание состояния предотвращает неоднозначность осевого положения.
- **Бит 0:** указывают позицию точки пересечения ручных осей (A4, A5, A6).

Позиция	Значение
Перевернутое положение Робот находится в перевернутом положении, если значение x точки пересечения ручных осей, относительно системы координат A1, является отрицательным.	Бит 0 = 1
Основная зона Робот находится в основной зоне, если значение x точки пересечения ручных осей, относительно системы координат A1, является положительным.	Бит 0 = 0

Система координат A1 идентична системе координат \$ROBROOT, если ось 1 стоит на 0°. При значениях, неравных 0°, выполняется перемещение вместе с осью 1.



**Рис. 16-9: Пример: Точка пересечения ручных осей (красная точка) находится в основной зоне.**

- **Бит 1:** указывает позицию оси 3. Угол, при котором значение бита 1 изменяется, зависит от типа робота.

Для роботов, чьи оси 3 и 4 пересекаются, действует следующее:

Позиция	Значение
$A3 \geq 0^\circ$	Бит 1 = 1
$A3 < 0^\circ$	Бит 1 = 0

В роботах со смещением между осью 3 и 4 угол, при котором значение бита 1 изменяется, зависит от величины этого смещения.

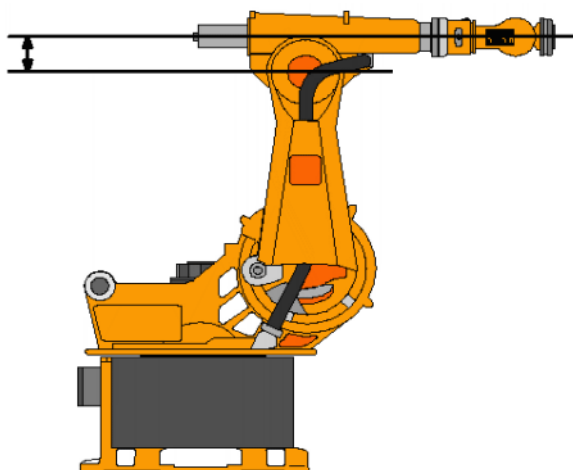


Рис. 16-10: Смещение между A3 и A4: пример KR 30

- **Бит 2:** указывает позицию оси 5.

Позиция	Значение
$A5 > 0$	Бит 2 = 1
$A5 \leq 0$	Бит 2 = 0

- **Бит 3 не используется** и всегда равен 0.
- **Бит 4:** указывает, была ли точка заучена с помощью прецизионного робота или нет.

Точка, вне зависимости от значения бита, может быть пройдена как прецизионным, так и непрецизионным роботом. Бит 4 служит лишь для справки и не влияет на то, как система управления роботом рассчитывает точку. Это означает, что, если робот программируется в автономном режиме, битом 4 можно пренебречь.

Описание	Значение
Точка не была заучена с помощью прецизионного робота.	Бит 4 = 0
Точка была заучена с помощью прецизионного робота.	Бит 4 = 1

**ПОВОРОТ**

- Указание поворота позволяет осуществлять перемещение в осевые углы больше  $+180^\circ$  или меньше  $-180^\circ$  без особой стратегии перемещения (например, промежуточных точек). Отдельные биты при ротационных осях определяют знак осевого значения следующим образом:

Бит = 0: угол  $\geq 0^\circ$   
 Бит = 1: угол  $< 0^\circ$

- Обзор всех осей

Значение	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0
0	$A6 \geq 0^\circ$	$A5 \geq 0^\circ$	$A4 \geq 0^\circ$	$A3 \geq 0^\circ$	$A2 \geq 0^\circ$	$A1 \geq 0^\circ$
1	$A6 < 0^\circ$	$A5 < 0^\circ$	$A4 < 0^\circ$	$A3 < 0^\circ$	$A2 < 0^\circ$	$A1 < 0^\circ$

- Пример

```
DECL POS XP1 = {X 900, Y 0, Z 800, A 0, B 0, C 0, S 6, T 19}
```

T 19 соответствует T 'B010011'. Это означает:

Ось	Угол	Двоичный
A 1	отрицательный	1
A 2	отрицательный	2
A 3	положительный	4
A 4	положительный	8
A 5	отрицательный	16
A 6	положительный	32

### Порядок действий

1. Выбрав уровень эксперта, загрузите программу в редактор нажатием на кнопку **Открыть**
2. Выполните манипуляции со значениями **состояние** и **поворот**. Промежуточно сохраните вновь рассчитанные значения в новой переменной
3. Проверьте предварительные настройки для программирования перемещения и примите их или выполните повторную инициализацию:
  - Инструмент (\$TOOL и \$LOAD)
  - Настройки базы (\$BASE)
  - Управляемый роботом или внешний инструмент (\$IPO\_MODE)
  - Скорость
  - Ускорение
  - При необходимости расстояние сглаживания
  - При необходимости контроль ориентации
4. Создайте команду перемещения из следующих компонентов:
  - Вид перемещения (PTP, LIN, CIRC)
  - Целевая точка (при CIRC также вспомогательная точка)
  - При CIRC угол круга (CA)
  - Активируйте сглаживание (C\_PTP, C\_DIS, C\_ORI, C\_VEL)
5. При повторном перемещении назад к пункту 3
6. Закройте редактор и сохраните



## 17 Работа с системой управления верхнего уровня

### 17.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Подготовка к запуску программы с ПЛК
- Настройка соединения с ПЛК
- Конфигурирование интерфейса «Внешняя автоматика»



Рис. 17-1: Смена главы

### 17.2 Подготовка к запуску программы от ПЛК

Продолжи-  
тельность: 00:30:00

Оснащение:

Информация:

Сертификат:

#### Робот в сочетании с установкой

Если процессами робота должен управлять центральный пункт (управляющий компьютер или ПЛК), то это осуществляется по интерфейсу «Внешняя автоматика».



Рис. 17-2: Соединение с ПЛК




**Принцип  
структуры  
установки**

Чтобы установить связь между KR C4 и ПЛК, требуется следующее:


- **режим работы «Внешняя автоматика»:** режим работы, в котором главный компьютер или ПЛК берет на себя управление системой робота;
- **CELL.SRC:** организационная программа для выбора внешней программы робота;
- **обмен сигналами между ПЛК и роботом: интерфейс «Внешняя автоматика»** для конфигурирования входных и выходных сигналов:
  - управляющие сигналы к роботу (входы): сигнал запуска и останова, номер программы, квитирование ошибки;
  - состояние робота (выходы): состояние приводов, положения, ошибки и т. д.

**Указания по  
технике безопас-  
ности для  
внешнего  
запуска  
программы**

После выбора программы CELL следует выполнить перемещение SAK.

 **ОСТОРОЖНО** Перемещение SAK выполняется в качестве перемещения PTP от фактического положения к целевому положению, если выбранная запись перемещений содержит команду перемещения PTP. Если выбранная запись перемещений содержит LIN или CIRC, то перемещение SAK выполняется в качестве перемещения LIN. Наблюдать за перемещением во избежание столкновений. При перемещении SAK скорость автоматически снижается.

Если перемещение SAK уже было выполнено, то при внешнем запуске перемещение SAK больше не будет выполняться.

 **ОСТОРОЖНО** В режиме внешней автоматики ход SAK отсутствует. Это значит, что робот после запуска выходит на первое запрограммированное положение с за запрограммированной (не сниженной) скоростью и там не останавливается.

**Порядок  
действий для  
внешнего  
запуска  
программы****Необходимые условия**

- Выбран режим работы T1 или T2.
  - Входы и выходы для внешней автоматики и программы CELL.SRC сконфигурированы.
1. В навигаторе выбрать программу CELL.SRC. Программа CELL всегда находится в директории KRC:\R1.
  2. Установить коррекцию программы на 100 %. (Это рекомендуемая настройка. При необходимости можно настроить другое значение).

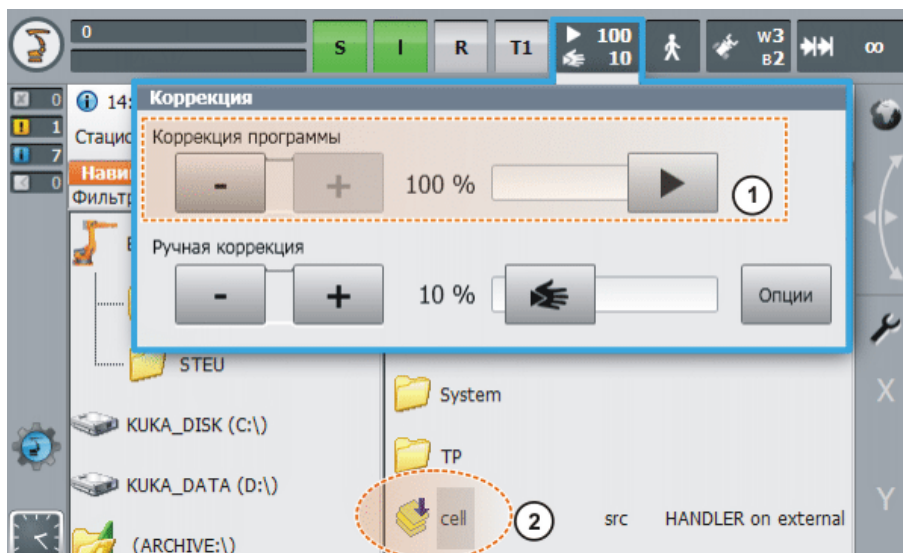


Рис. 17-3: Выбор Cell и установка ручной коррекции

- 1 Настройка NOV
- 2 Выбор Cell.src
3. Выполнить перемещение SAK:  
нажать и удерживать переключатель подтверждения. Затем нажать и удерживать клавишу запуска, пока в окне сообщений не появится сообщение «SAK достигнут».
4. Выбрать режим работы «Внешняя автоматика».
5. Запустить программу через систему управления верхнего уровня (ПЛК).

### 17.3 Настройка соединения с ПЛК (Cell.src)

---

Продолжительность: 00:20:00  
 Оснащение: Офисный ПК  
 Информация:  
 Сертификат:

---

#### Организационная программа Cell.src

Для управления передаваемыми от ПЛК номерами программ используется организационная программа Cell.src. Она всегда находится в папке R1. Как и любую обыкновенную программу, программу Cell также можно индивидуально настроить, однако при этом должна сохраниться структура программы.

Структура и функциональность программы Cell

```

1 DEF CELL ( )
6 INIT
7 BASISTECH INI
8 CHECK HOME
9 PTP HOME Vel= 100 % DEFAULT
10 AUTOEXT INI
11 LOOP
12 P00 (#EXT_PGNO,#PGNO_GET,DMY[1,0] )
13 SWITCH PGNO ; Select with Programnumber
14
15 CASE 1
16 P00 (#EXT_PGNO,#PGNO_ACKN,DMY[],0 )
17 ;EXAMPLE1 ( ) ; Call User-Program
18
19 CASE 2
20 P00 (#EXT_PGNO,#PGNO_ACKN,DMY[],0 )
21 ;EXAMPLE2 ( ) ; Call User-Program
22
23 CASE 3
24 P00 (#EXT_PGNO,#PGNO_ACKN,DMY[],0 )
25 ;EXAMPLE3 ( ) ; Call User-Program
26
27 DEFAULT
28 P00 (#EXT_PGNO,#PGNO_FAULT,DMY[],0 )
29 ENDSWITCH
30 ENDLLOOP
31 END

```

Рис. 17-4: Программа Cell

- 1 Инициализация и положение Home:
  - инициализация базовых параметров;
  - проверка положения робота после принятия положения Home;
  - инициализация интерфейса внешней автоматки.
- 2

Непрерывный цикл:

  - запрос номера программы модулем «P00»;
  - переход в цикл выбора с определенным номером программы.
- 3 Цикл выбора номера программы:
  - соответственно номеру программы (хранится в переменной «PGNO») выполняется переход к текущей ветви («CASE»);
  - затем обрабатывается введенная в ветвь программа робота;
  - недействительные номера программы приводят к переходу в ветвь «Default»;
  - после успешной обработки цикл повторяется.

Порядок действий

1. Переключиться в группу пользователей «Эксперты».
2. Открыть программу CELL.SRC.
3. В разделах CASE заменить обозначение EXAMPLE именем программы, которую должна вызвать текущая программа с номером. Удалить точку с запятой перед названием.

```

LOOP
  P00 (#EXT_PGNO,#PGNO_GET,DMY[],0 )
  SWITCH PGNO ; Select with Programnumber

  CASE 1
    P00 (#EXT_PGNO,#PGNO_ACKN,DMY[],0 )
    main()

  CASE 2
    P00 (#EXT_PGNO,#PGNO_ACKN,DMY[],0 )
    body_38()
    body_515()

  DEFAULT
    P00 (#EXT_PGNO,#PGNO_FAULT,DMY[],0 )
  ENDSWITCH
ENDLOOP

```

Рис. 17-5: Пример отредактированной программы Cell

4. Закрыть программу и сохранить изменения.

## 17.4 Конфигурирование и применение режима «Внешняя автоматика»

### Описание



Рис. 17-6: Соединение с ПЛК

- При использовании интерфейса «Внешняя автоматика» процессами робота управляет вышестоящая система управления (напр., ПЛК).
- Через интерфейс «Внешняя автоматика» вышестоящая система управления передает в систему управления роботом сигналы для процессов робота (напр., разрешение на перемещение, квитирование ошибки, запуск программы и т. д.). Система управления роботом передает в вышестоящую систему управления сведения о рабочих состояниях и сбоях.

Чтобы использовать интерфейс «Внешняя автоматика», необходимо выполнить следующие конфигурации:

1. Конфигурацию программы CELL.SRC.
2. Конфигурацию входов/выходов интерфейса внешней автоматки.

**Использование входов/выходов интерфейса «Внешняя автоматика»**

Обзор основных сигналов интерфейса

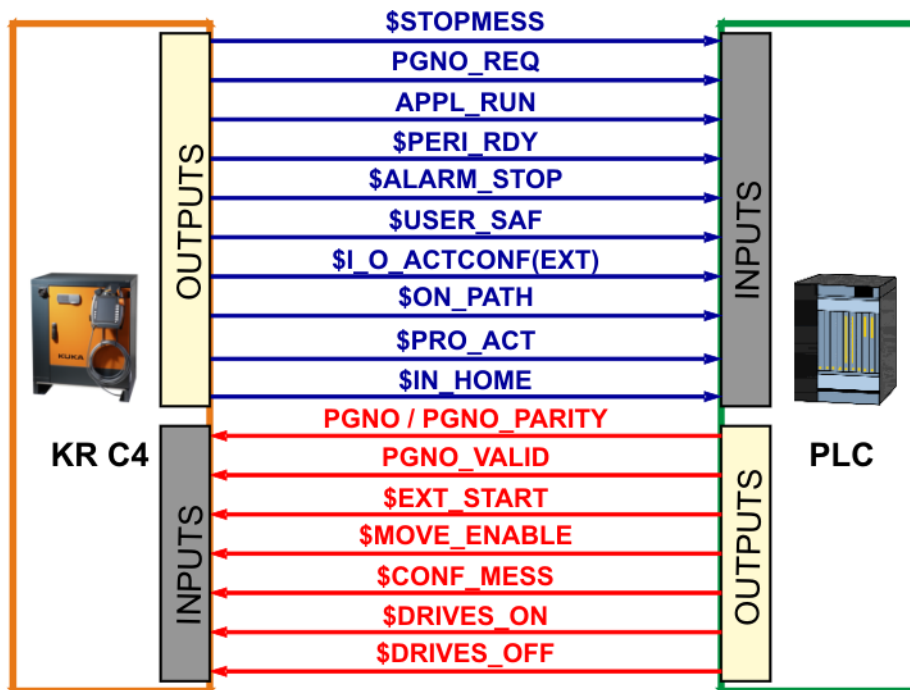


Рис. 17-7: Обзор основных сигналов внешней автоматике

Входы (с точки зрения системы управления роботом)

■ **PGNO\_TYPE** - тип номера программы

Эта переменная устанавливает, в каком формате записывается номер программы, переданный вышестоящей системой управления.

Значение	Описание	Пример
1	Запись в качестве двоичного числа. Номер программы передается вышестоящей системой управления в виде целочисленного значения с двоичной кодировкой.	0 0 1 0 0 1 1 1 => PGNO = 39
2	Запись в качестве значения BCD. Номер программы передается вышестоящей системой управления в виде десятичного значения с двоичной кодировкой.	0 0 1 0 0 1 1 1 => PGNO = 27
3	Запись как «1 из N»*. Номер программы передается вышестоящей системой управления или периферией в виде значения с кодировкой «1 из N».	0 0 0 0 0 0 0 1 => PGNO = 1 0 0 0 0 1 0 0 0 => PGNO = 4

\* В этом формате передачи значения PGNO\_REQ, PGNO\_PARITY и PGNO\_VALID не обрабатываются, поэтому не играют роли.

■ **PGNO\_LENGTH** - длина номера программы

Эта переменная устанавливает разрядность номера программы, переданного вышестоящей системой управления. Диапазон значений: 1 - 16

Если PGNO\_TYPE имеет значение 2, то допущены только разрядности 4, 8, 12 и 16.

- **PGNO\_PARITY** - бит четности номера программы

Вход, на который вышестоящая система управления передает бит четности.

Контроль четности служит для распознавания слов данных, переданных с ошибкой. Словом данных здесь обозначается последовательность битов. «Четность» – это количество битов со значением 1 в слове данных. Четность (англ. «even») соблюдается, если количество таких битов четное; в ином случае устанавливается состояние нечетности (англ. «odd»). Кодировка контроля четности добавляет к слову данных бит контроля четности (parity bit). Это происходит таким образом, что все подлежащие передаче кодовые слова имеют одинаковое состояние четности/нечетности. В зависимости от того, передаются данные с четностью или нечетностью, добавляется «Е» или «О» (см. примеры в графических изображениях). Результат, на один бит длиннее слова данных, обозначается здесь как кодовое слово. Метод распознавания ошибок с использованием битов четности называется проверкой четности. Место возникновения ошибки в пределах кодового слова неизвестно, поэтому коррекция ошибок невозможна. Кроме того, у бита четности ( $N = 1$ ) может быть зарегистрировано только нечетное количество ошибок битов в кодовом слове. Четное количество ошибок битов не регистрируется, т.к. в кодовом слове состояние четности остается неизменным.

Вход	Функция
Отрицательное значение	Нечетность
0	Оценка не выполняется
Положительное значение	Четность

**Пример 1 (четность Even):** если для передачи данных установлена четность Even (четная сумма → бит четности: 0, нечетная сумма → бит четности: 1), то для обоих последующих примеров действительно: слово данных 0011.1010 содержит четыре единицы. Четыре – четное число, бит контроля четности определен нулем, результирующее кодовое слово 0011.1010 0. В свою очередь, слово данных 1010.0100 имеет нечетную сумму четности и кодируется в кодовое слово 1010.0100 1.

**Пример 2 (четность Odd):** если для передачи данных установлена четность Odd (четная сумма → бит четности: 1, нечетная сумма → бит четности: 0), то для обоих последующих примеров действительно: слово данных 0011.1010 содержит четыре единицы. Четыре – четное число, бит контроля четности определен единицей, результирующее кодовое слово 0011.1010 1. В свою очередь, слово данных 1010.0100 имеет нечетную сумму четности и кодируется в кодовое слово 1010.0100 0.

Если PGNO\_TYPE имеет значение 3, то PGNO\_PARITY не оценивается.

- **PGNO\_VALID** - действительный номер программы


Вход, на который вышестоящая система управления передает команду на запись номера программы.


Вход	Функция
Отрицательное значение	Номер принимается с нисходящим фронтом сигнала.

Вход	Функция
0	Номер принимается с нарастающим фронтом сигнала на линии EXT_START.
Положительное значение	Номер принимается с нарастающим фронтом сигнала.

■ **\$EXT\_START** - внешний пуск

Включением этого выхода при активном интерфейсе ввода/вывода может быть запущена или продолжена программа (обычно CELL.SRC).


 Оценивается только нарастающий фронт сигнала.


 **ОСТОРОЖНО** В режиме внешней автоматики ход SAK отсутствует. Это значит, что робот после запуска выходит на первое запрограммированное положение с за программированной (не сниженной) скоростью и там не останавливается.

■ **\$MOVE\_ENABLE** - деблокировка перемещения

Этот вход используется для контроля приводов робота вышестоящей системой управления.


Сигнал	Функция
TRUE	Разрешено перемещение вручную и выполнение программы
FALSE	Остановка всех приводов и блокировка всех активных команд

 Если приводы остановлены вышестоящей системой управления, выводится сообщение «Разрешение общего перемещения». Перемещение робота возможно только после удаления этого сообщения и поступления повторного внешнего сигнала запуска.

 Во время пуска в эксплуатацию переменная \$MOVE\_ENABLE часто проектируется на значение \$IN[1025]. Если после этого забыть спроектировать другой вход, то внешний пуск невозможен.

■ **\$CONF\_MESS** – квитирование сообщений

При активации этого входа вышестоящая система управления самостоятельно квитировывает сообщения об ошибках после устранения их причины.

 Оценивается только нарастающий фронт сигнала.

■ **\$DRIVES\_ON** - приводы вкл.

Если на этот вход подается импульс High длительностью не менее 20 мс, вышестоящая система управления включает приводы робота.

■ **\$DRIVES\_OFF** - приводы выкл.

Если на этот вход подается импульс Low длительностью не менее 20 мс, вышестоящая система управления выключает приводы робота.

Выходы (с точки зрения системы управления роботом)

■ **\$ALARM\_STOP** - АВАРИЙНЫЙ ОСТАНОВ



Этот выход сбрасывается в следующих ситуациях АВАРИЙНОГО ОСТАНОВА:

- нажатие кнопки АВАРИЙНОГО ОСТАНОВА на КСР (Внутр. авар. выкл.);
- Внешний АВАРИЙНЫЙ ОСТАНОВ



После АВАРИЙНОГО ОСТАНОВА тип этого события можно распознать по состояниям выходов **\$ALARM\_STOP** и **Внутр. авар. выкл.**:

- оба выхода установлены на FALSE: АВАРИЙНЫЙ ОСТАНОВ активирован на КСР;
  - **\$ALARM\_STOP** на FALSE, **Внутр. авар. выкл.** на TRUE: Внешний АВАРИЙНЫЙ ОСТАНОВ
- **\$USER\_SAF** - система защиты оператора/защитные двери  
Этот выход сбрасывается при срабатывании контрольного выключателя защитной решетки (режим EXT) или при отпускании переключателя предупреждения (режим работы T1 или T2).
  - **\$PERI\_RDY** - приводы готовы  
Активацией этого выхода система управления роботом сообщает вышестоящей системе управления, что приводы робота включены.
  - **\$STOPMESS** - извещения об останове  
Этот выход активируется системой управления роботом, чтобы указать вышестоящей системе управления на поступление сообщения, вызвавшего необходимость остановки робота. (Примеры: АВАРИЙНЫЙ ОСТАНОВ, деблокировка перемещения или защита оператора)
  - **\$I\_O\_ACTCONF** - активирована внешняя автоматика  
Этот выход устанавливается на TRUE, если выбран режим «Внешняя автоматика» и вход имеет значение \$I\_O\_ACT TRUE (обычно всегда \$IN[1025]).
  - **\$PRO\_ACT** - программа активируется/выполняется  
Этот выход всегда активирован в том случае, если выполняется какой-либо процесс на уровне робота. Процесс активирован, пока обрабатывается программа или прерывание. Выполнение программы заканчивается в конце программы только в случае, если все импульсные выходы и триггеры обработаны.

В ситуациях с остановом из-за ошибки следует различать следующие возможности:

- Если прерывания были активированы, но не обработаны на момент останова из-за ошибки, процесс считается не активированным (\$PRO\_ACT=FALSE).
- Если прерывания были активированы и обработаны на момент останова из-за ошибки, процесс считается активированным (\$PRO\_ACT=TRUE) до тех пор, пока программа прерывания не будет выполнена или не подойдет к останову (\$PRO\_ACT=FALSE).
- Если прерывания были активированы и программа подходит к останову, процесс считается не активированным (\$PRO\_ACT=FALSE). Если после этого момента выполняется условие прерывания, процесс считается активированным (\$PRO\_ACT=TRUE) до тех пор, пока программа прерывания не будет выполнена или не подойдет к останову (\$PRO\_ACT=FALSE).



- **PGNO\_REQ** - запрос номера программы

Смена сигнала на этом выходе – это требование к вышестоящей системе управления определить номер программы.

Если PGNO\_TYPE имеет значение 3, то PGNO\_REQ не оценивается.
- **APPL\_RUN** - выполняется прикладная программа

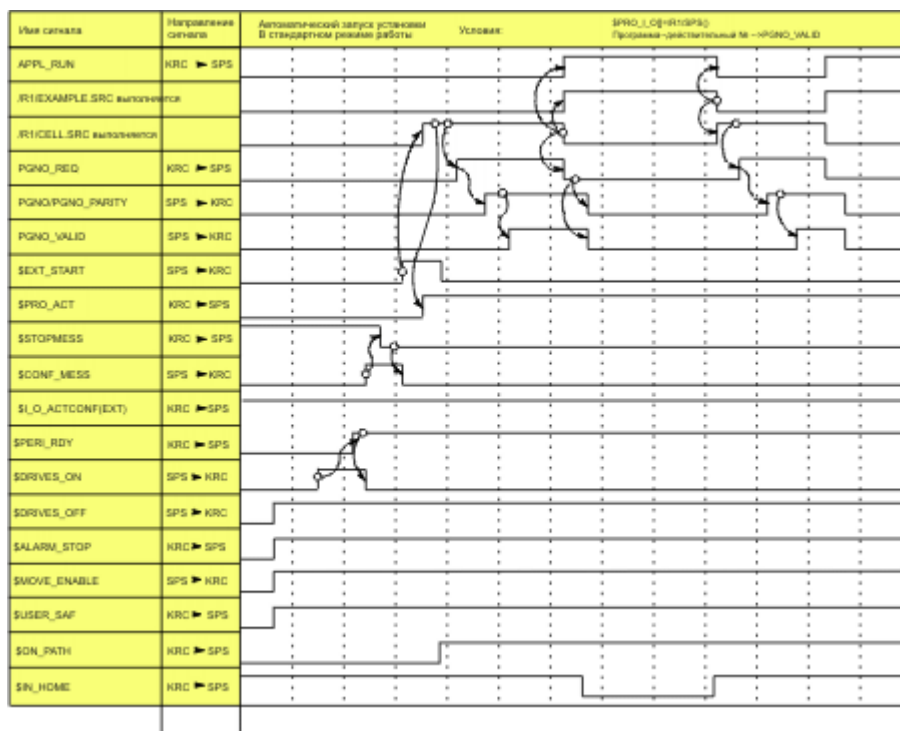
Активацией этого выхода система управления роботом сообщает вышестоящей системе управления, что в настоящее время выполняется программа.
- **\$IN\_HOME** - робот в позиции HOME

Этот выход сообщает вышестоящей системе управления, находится ли робот в позиции HOME.
- **\$ON\_PATH** - робот на траектории

Этот выход активирован все время, пока робот находится на своей запрограммированной траектории. После перемещения SAK активизируется выход ON\_PATH. Он остается активированным до тех пор, пока робот не покинет траекторию, программа не сбросится или не будет выполнен выбор строки. Сигнал ON\_PATH не имеет окна допуска; если робот покидает траекторию, этот сигнал сбрасывается.

**Принцип коммуникации в режиме внешней автоматике**

Обзор общего процесса



**Рис. 17-8: Автоматический запуск установки и стандартный режим работы с квитирование номера программы посредством PGNO\_VALID**

Разделение на участки

1. Включение приводов
2. Квитирование сообщений
3. Запуск программы Cell
4. Передача номера программы и выполнение приложения

Для каждого данного участка необходимо соблюдать соответствующие условия, а также обеспечить возможность передачи информации о состояниях робота на ПЛК.



Рис. 17-9: Квитирование сигналов

Целесообразно использовать эти заданные квитирования.

#### Включение приводов

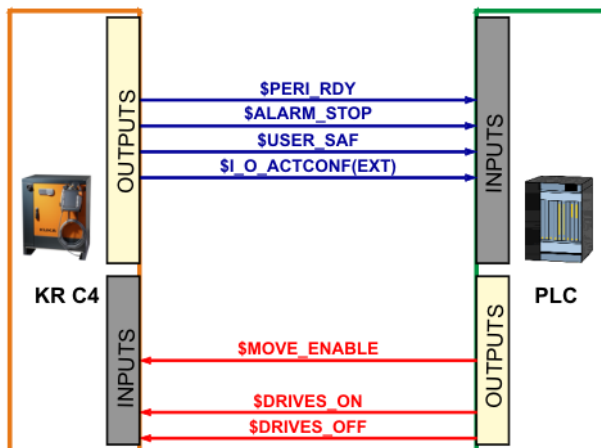


Рис. 17-12

- Условия
  - \$USER\_SAF - защитные двери закрыты
  - \$ALARM\_STOP - АВАРИЙНЫЙ ОСТАНОВ отсутствует
  - \$I\_O\_ACTCONF - активирована внешняя автоматика
  - \$MOVE\_ENABLE - деблокировка перемещения выполнена
  - \$DRIVER\_OFF - «приводы выкл.» не активировано
- Включение приводов
  - \$DRIVES\_ON - включение приводов мин. на 20 мс
- Приводы готовы
  - \$PERI\_RDY - при поступлении обратного сообщения для приводов сигнал \$DRIVES\_ON снимается

#### Квитирование сообщений

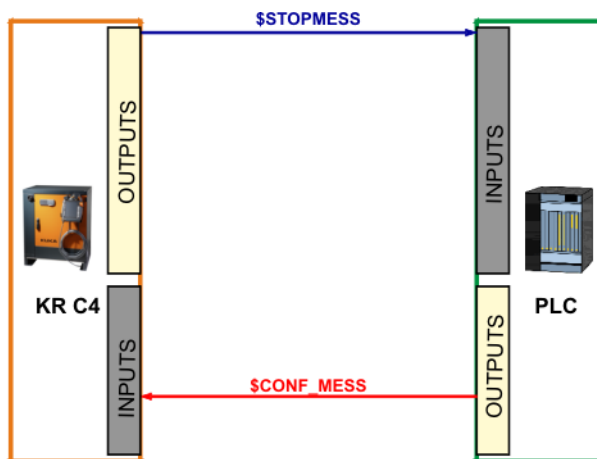


Рис. 17-16

- Условия
  - \$STOPMESS - поступило извещение об останове
- Квитировать сообщение.
  - \$CONF\_MESS - квитирование сообщения
- Квитируемые сообщения удалены
  - \$STOPMESS - извещение об останове отменено, \$CONF\_MESS теперь может быть снято

**Запуск программы (CELL.SRC) извне**

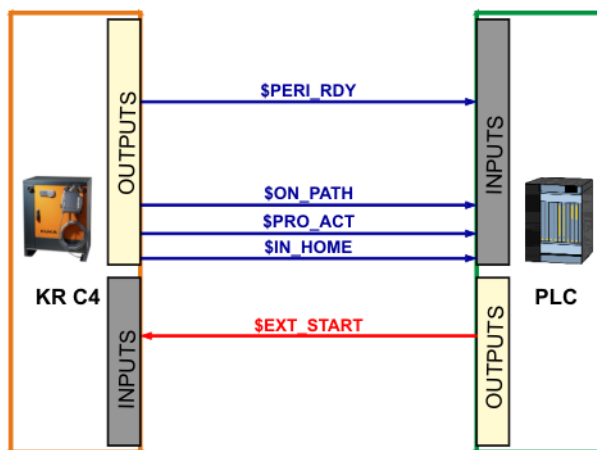


Рис. 17-21

- Условия
  - \$PERI\_RDY - приводы готовы
  - \$IN\_HOME - робот в позиции HOME
  - Нет \$STOPMESS - извещения об останове отсутствуют
- Внешний пуск
  - \$EXT\_START- включение внешнего пуска (положительный фронт)
- Программа CELL выполняется
  - \$PRO\_ACT - извещение о выполнении программы CELL
  - \$ON\_PATH - как только робот оказывается на своей запрограммированной траектории, сигнал \$EXT\_START снимается

**Передача номера программы и выполнение прикладной программы**

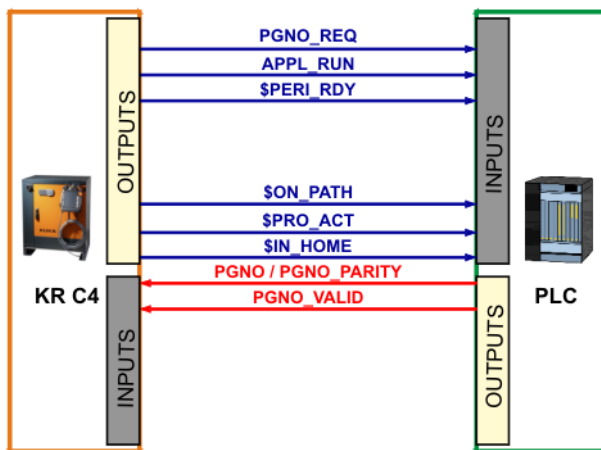


Рис. 17-28

- Условия
  - \$PERI\_RDY - приводы готовы
  - \$PRO\_ACT - программа CELL выполняется
  - \$ON\_PATH - робот на траектории
  - \$IN\_HOME - робот в позиции HOME, при повторном пуске не требуется
  - PGNO\_REQ - поступил запрос номера программы
- Передача номера программы и подтверждение
  - Передача номера программы (правильный тип данных (PGNO\_TYPE), длина номера программы (PGNO\_LENGTH) и первый бит номера программы (PGNO\_FBIT) настроены)
  - PGNO\_VALID- подтверждение номера программы (положительный фронт)
- Выполняется прикладная программа
  - APPL\_RUN - извещение о выполнении прикладной программы
  - Робот покидает позицию HOME. После завершения прикладной программы робот возвращается в позицию HOME.

#### Порядок действий

1. В главном меню выбрать пункт **Конфигурация > Входы/выходы > Внешняя автоматика**.
2. В столбце **Значение** выделить ячейку, которая должна быть обработана, и нажать на **Правка**.
3. Указать нужное значение и сохранить нажатием **ОК**.
4. Повторить шаги 2 и 3 для всех редактируемых значений.
5. Закрыть окно. Изменения принимаются системой.

1	2	3	4	5
Внешняя автоматика – конфигурация: Входы				
	Обозначение	Тип	Имя	Значение
1	Тип, № программы	Var	PGNO_TYPE	1
2	Отражение номеров программ	Var	REFLECT_PROG_I	0
3	Ширина бита, № программы	Var	PGNO_LENGTH	8
4	Первый бит, номер программы	NO	PGNO_FBIT	33
5	Бит контроля четности	NO	PGNO_PARITY	41
6	№ программы действителен	NO	PGNO_VALID	42
7	Запуск программы	NO	\$EXT_START	1026
8	Деблокировка перемещения	NO	\$MOVE_ENABLE	1025
9	Подтверждение ошибки	NO	\$CONF_MESS	1026
10	Приводы выключены (инверсия)	NO	\$DRIVES_OFF	1025
11	Приводы включены	NO	\$DRIVES_ON	140
12	Запустить интерфейс	NO	\$I_O_ACT	1025

Рис. 17-31: Конфигурация внешней автоматки: входы

Поз.	Описание
1	Номер
2	Полное имя входа/выхода
3	Тип <ul style="list-style-type: none"> <li>■ <b>Зеленый:</b> вход/выход</li> <li>■ <b>Желтый:</b> переменная или системная переменная (\$...)</li> </ul>
4	Имя сигнала или переменной
5	Номер входа/выхода или номер канала
6	Выходы тематически распределены по вкладкам.

	Обозначение	Тип	Имя	Значение
1	Система управления готова	<i>AKO</i>	\$SRC_RDY1	137
2	Контур аварийного выключения замкнут	<i>AKO</i>	\$ALARM_STOP	1013
3	Переключатель защиты оператора закрыт	<i>AKO</i>	\$USER_SAF	1011
4	Приводы готовы	<i>AKO</i>	\$PERI_RDY	1012
5	Робот юстирован	<i>AKO</i>	\$ROB_CAL	1001
6	Интерфейс активен	<i>AKO</i>	\$I_O_ACTCONF	140
7	Общая ошибка	<i>AKO</i>	\$STOPMESS	1010
8	Первый бит для создания отражения программы	<i>AKO</i>	PGNO_FBIT_REFL	999
9	Внутренний аварийный останов	<i>AKO</i>	\$ALARM_STOP_I	1002

Условия запуска | Состояние программы | Позиция робота | Режим ра

Рис. 17-32: Конфигурация внешней автоматики – выходы



## 18 Программирование с помощью WorkVisual

### 18.1 Обзор

В этом учебном разделе рассматриваются следующие темы:

- Описание изделия
- Подсоединение к системе управления роботом
- Управление проектами
- Программирование с WorkVisual



Рис. 18-1: Смена главы

Программный пакет **WorkVisual** представляет собой прикладное окружение для работы с робототехническими ячейками, которыми управляет KR C4. Он имеет следующие функции:

- Структурирование и соединение магистральных шин
- Автономное программирование роботов
- Конфигурировать комплекс роботов офлайн
- редактирование конфигурации системы безопасности
- Передача проектов в систему управления роботом
- Передача проектов из системы управления роботом в WorkVisual
- Сравнение проекта с другим проектом и принятие различий при необходимости
- Управление длинными текстами
- Управление пакетами опций
- Функция диагностики
- Просмотр системных данных управления роботом онлайн
- Конфигурирование трассировки, начало записи, анализ трассировки (с помощью осциллоскопа)

### 18.2 Соединение с системой WorkVisual

Предоставление IP-адреса вручную

Условия

- Шкаф управления работает без соединения к компьютерной сети.
- Наличие ноутбука, права администратора и свободный сетевой порт.
- Сетевой разъем в CSP подсоединен к внутреннему линейному интерфейсу KUKA компьютера KUKA (также возможно подсоединение через внутренний коммутатор).

**УВЕДОМЛЕНИЕ** При стандартной поставке KLI предварительно настроен на статический IP-адрес  
172.31.1.147





Рис. 18-2: Ввод IP-адреса вручную

Предоставление IP-адреса шкафу управления вручную

- Конфигурирование интерфейса KLI с помощью HMI поддерживается вспомогательной программой.
- Требуется профиль пользователя «Эксперт».
- Ошибки в конфигурации отмечаются красным цветом и запретом на сохранение.
- Пользователю на выбор предлагается стандартная маска и детальная маска конфигурации.
- В стандартной маске можно настраивать только интерфейс Windows (подключение к офисной сети).
- В детальной маске можно выполнить конфигурацию дополнительных виртуальных сетей.
- Путь меню: клавиша KUKA > Ввод в эксплуатацию > Конфигурация сети.

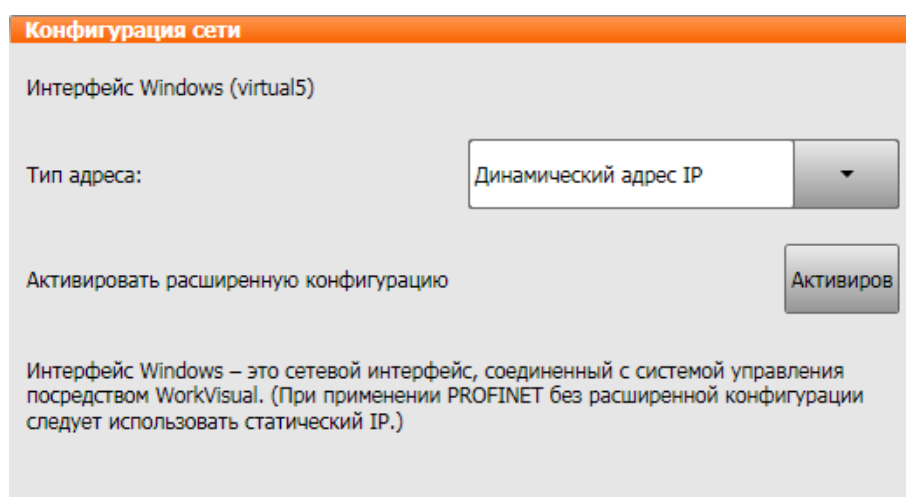


Рис. 18-3: KLI-HMI virtual5

Тип адреса	Значение
Динамический IP	Все настройки осуществляются с сервера DHCP
Постоянный IP	IP-адрес Маска подсети Стандартный межсетевой шлюз должен настраиваться отдельно
Без IP	Чтобы временно скрыть интерфейс
IP в реальном времени	Roboteam (комплекс роботов)
Смешанный IP	Специальная конфигурация технологических пакетов

**i** Если виртуальная сеть *virtual5* настроена на опцию **DHCP**, связь через **PROFINET** невозможна.

- Нажать кнопку **Активировать**.
- Появляется страница конфигурации с детальной маской.

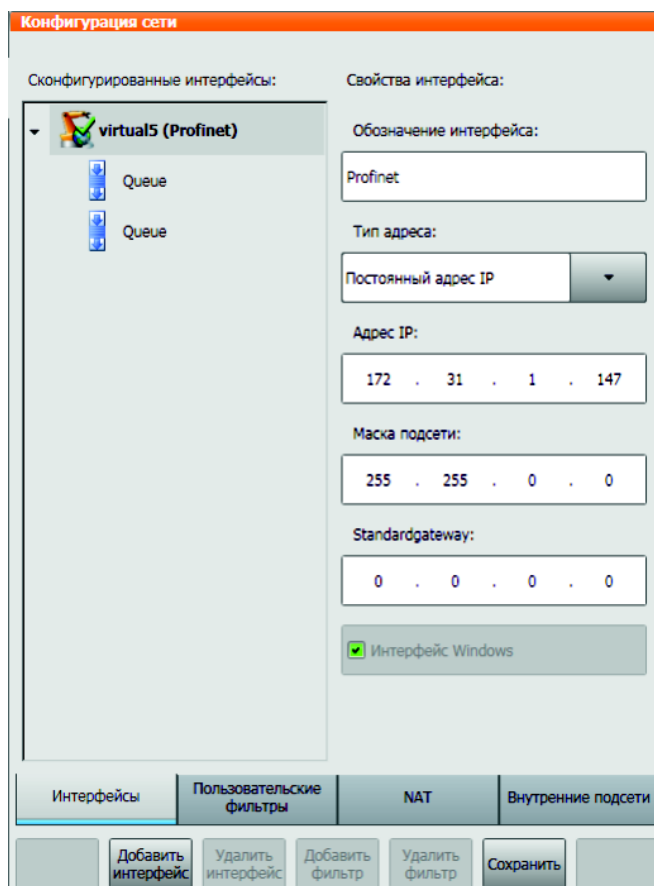


Рис. 18-4: Конфигурация сети: Virtual 5

Интерфейсы	Интерфейс	Значение
virtual5	Обозначение	Имя для интерфейса, например, PROFINET
	Тип адреса	Постоянный IP Динамический IP Без IP IP в реальном времени Смешанный IP
	IP-адрес	PROFINET-IP ввести вручную
	Маска подсети	Маску подсети ввести вручную
	Стандартный межсетевой шлюз	Стандартный межсетевой шлюз ввести вручную
	Интерфейс Windows	Указывает, действуют ли в отношении этого интерфейса правила NAT. Только при настроенном интерфейсе действуют по умолчанию.
1-я очередь	Приемный фильтр Profinet	Специальные ненастраиваемые порты PROFINET
2-я очередь	Пакеты приемного фильтра	Все принимают - все пакеты принимаются Цель IP-адреса – пакеты принимаются только для этого IP-адреса

- Кнопкой **Interfaces** можно добавить новый виртуальный интерфейс «**virtual6**».
- Это необходимо, если для PROFINET и офисной сети будут использоваться отдельные области IP-адресов.

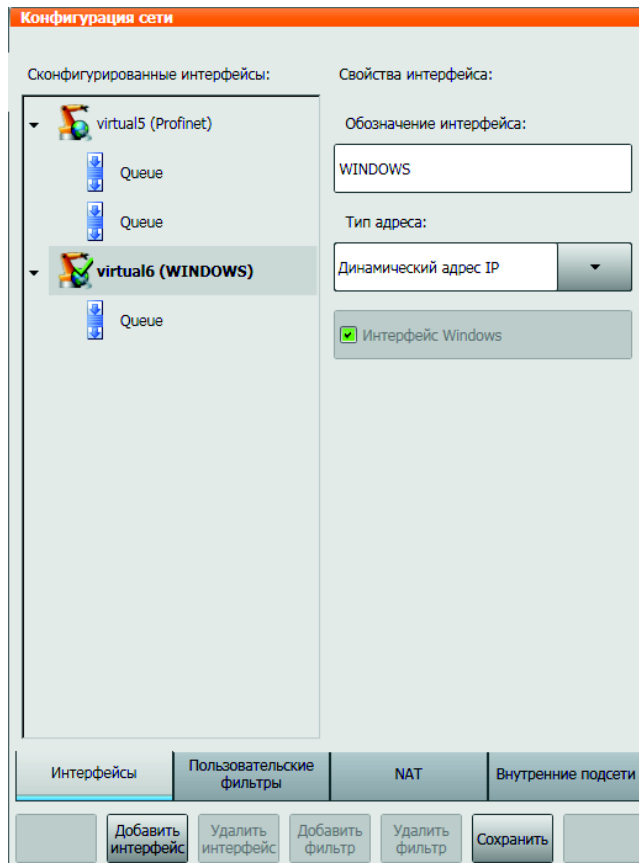


Рис. 18-5: Конфигурация сети: Virtual 6

Интерфейсы	Интерфейс	Значение
virtual6	Обозначение	Имя для интерфейса, например, «Офисная сеть»
	Тип адреса	Постоянный IP Динамический IP Без IP IP в реальном времени Смешанный IP
	IP-адрес	PROFINET-IP ввести вручную
	Маска подсети	Маску подсети ввести вручную
	Стандартный межсетевой шлюз	Стандартный межсетевой шлюз ввести вручную
	Интерфейс Windows	Указывает, действуют ли в отношении этого интерфейса правила NAT. При двух интерфейсах следует выбрать ручную.
Очередь	Пакеты приемного фильтра	Все принимают - все пакеты принимаются Цель IP-адреса – пакеты принимаются только для этого IP-адреса



В этом разделе курса в качестве примера описано предоставление IP-адреса ноутбуку с операционной системой Windows 7.

Предоставление IP-адреса ноутбуку вручную

1. В Windows вызвать **Управление системой > Сеть и Интернет > Центр сети и допусков**.
2. Вызвать опцию «Изменить настройки адаптера» (1).

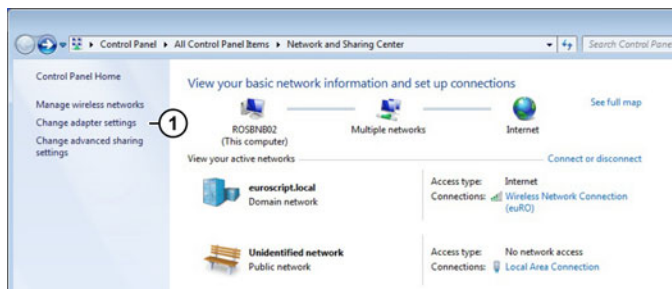


Рис. 18-6: Центр сети и допусков

3. Выделить курсором правильную карту LAN и открыть ее двойным щелчком курсора.

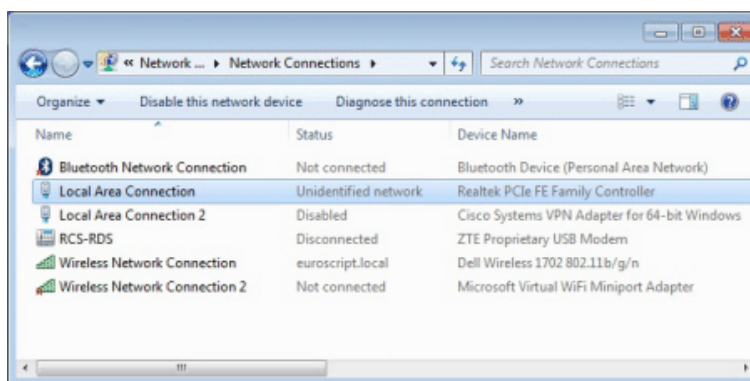


Рис. 18-7: Сетевые соединения

4. В окне **Статус соединения LAN** выбрать настройку **Свойства**.  
Для открытия данной настройки необходимы права администратора.

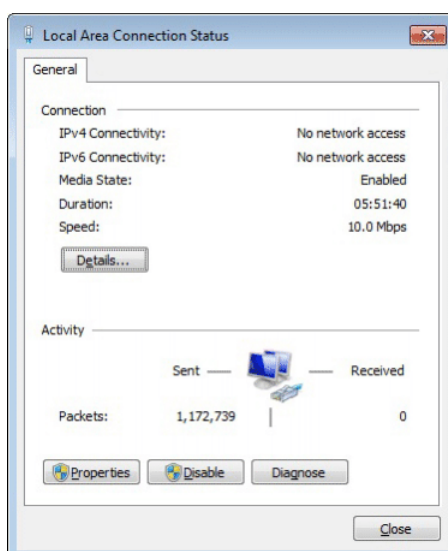


Рис. 18-8: Статус соединения LAN

5. Во вкладке «Сеть» выделить опцию **Интернет протокол версия 4 (TCP/IPv4)**.  
Открыть для выбранной опции вкладку **Свойства**.

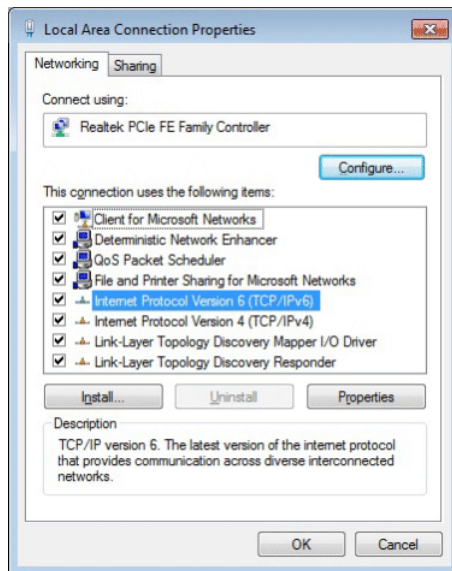


Рис. 18-9: Окно Свойства сетевого соединения

6. Щелкнуть курсором на кнопку **Свойства**.

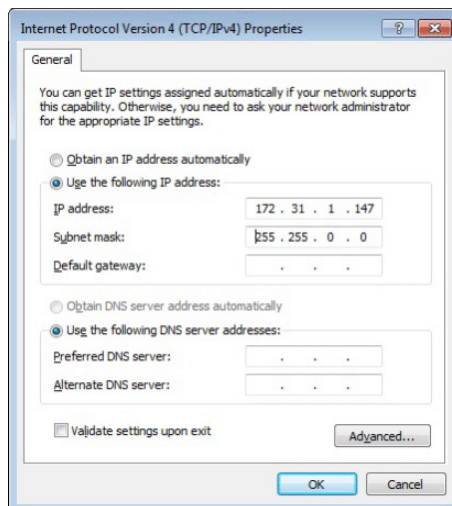


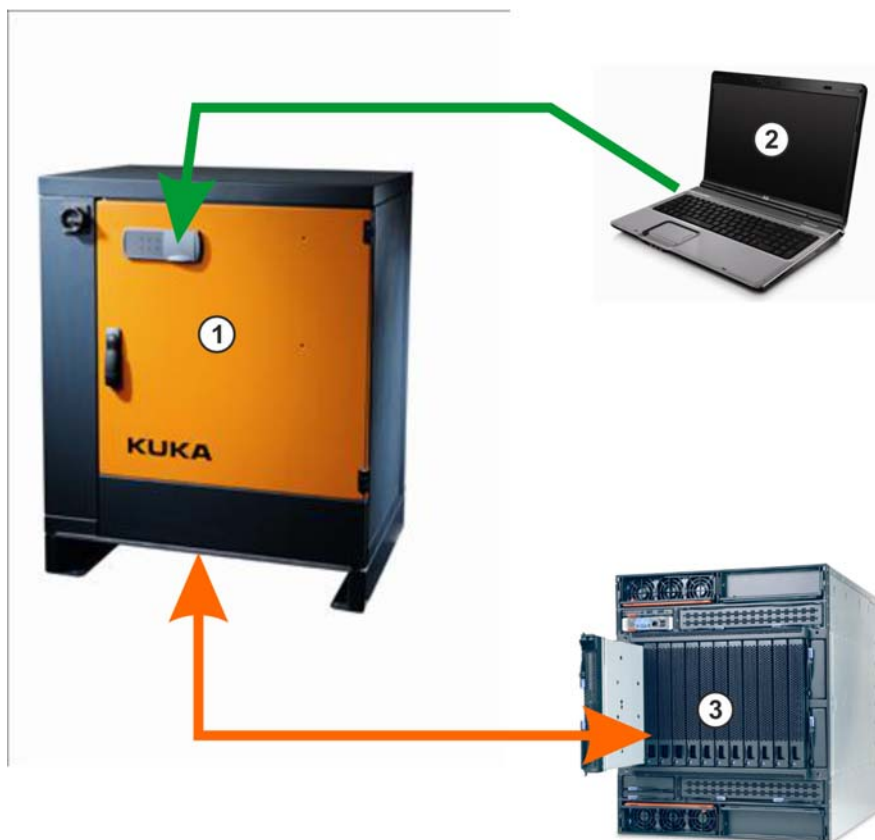
Рис. 18-10: Окно Свойства альтернативная настройка

7. Переключатель должен быть установлен на опцию **Определено пользователем**.
8. Внести IP-адрес и данные маски подсети.
9. Закрыть все окна кнопками **ОК**.

### Предоставление IP-адреса через DHCP

#### Условия

- Шкаф управления подсоединен к компьютерной сети.
- Наличие ноутбука, права администратора и свободный сетевой порт.
- Сетевой разъем в CSP подсоединен к внутреннему линейному интерфейсу KUKA компьютера KUKA (также возможно подсоединение через внутренний коммутатор).
- Автоматическое предоставление адреса выполняется через компьютерную систему (DHCP).



**Рис. 18-11: IP через DHCP**

Порядок действий

1. Соединить ноутбук со шкафом управления (>>> Рис. 18-11 ).
2. Если описанные выше условия выполнены, то шкафу управления и ноутбуку автоматически присваивается IP-адрес сервером компьютерной системы (3).
3. Проверить полученный IP-адрес шкафа управления на мониторе диагностики.

Меню «Клавиша KUKA» - «Диагностика» - «Монитор диагностики» - «Virtual5» или «Virtual 6»:



**Рис. 18-12: Монитор диагностики**

4. Проверить полученный IP-адрес ноутбука.
5. Открыть WorkVisual.

## 18.3 Управление проектом с помощью WorkVisual

### Выполнение проекта

1. Загрузка проекта из системы управления роботом в WorkVisual  
(>>> 18.3.1 "Открытие проекта с помощью WorkVisual" Страница 251)
2. Изменение проекта, например, с помощью программ KRL  
(>>> 18.4 "Редактирование программ KRL с помощью WorkVisual" Страница 266)
3. Сравнение проекта (слияние)  
(>>> 18.3.2 "Сравнение проектов с помощью WorkVisual" Страница 255)
4. Загрузка проекта WorkVisual в систему управления роботом (повторное открывание)  
(>>> 18.3.3 "Передача проекта в систему управления роботом (установка)" Страница 259)
5. Активация проекта (подтверждение)  
(>>> 18.3.4 "Активация проекта в системе управления роботом" Страница 263)

### 18.3.1 Открытие проекта с помощью WorkVisual

#### Краткое описание WorkVisual

Программный пакет **WorkVisual** представляет собой прикладное окружение для работы с робототехническими ячейками, которыми управляет KR C4. Он имеет следующие функции:

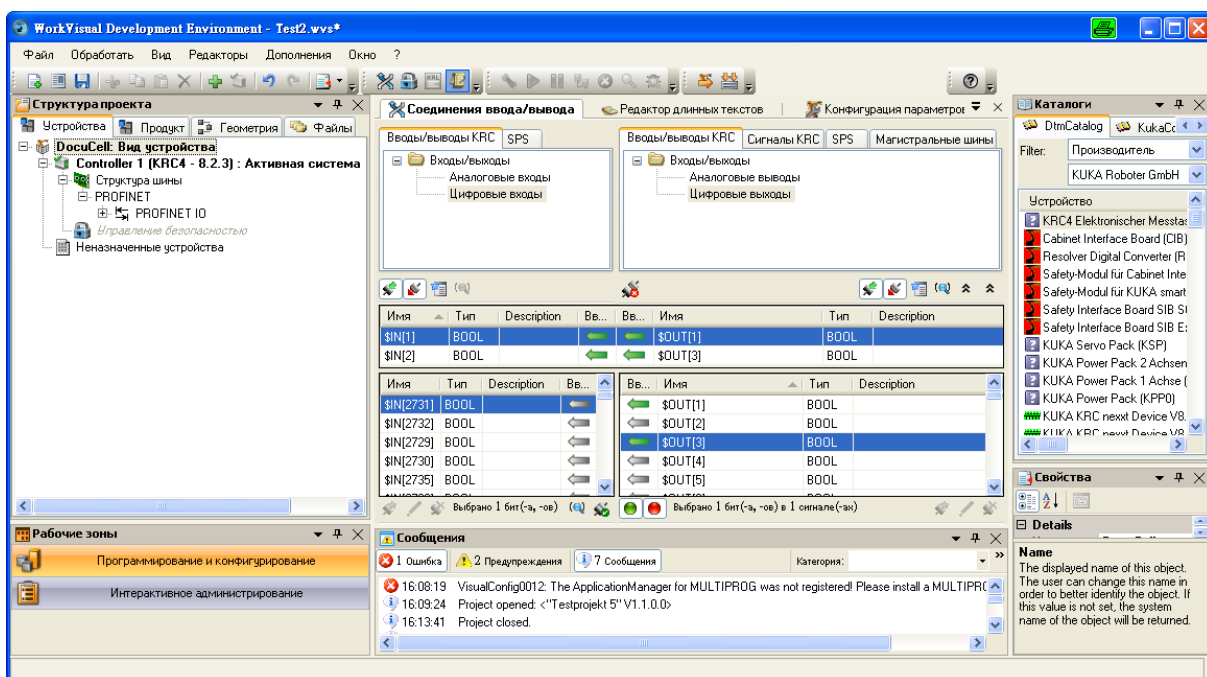


Рис. 18-13: Панель управления WorkVisual

- Передача проектов из системы управления роботом в WorkVisual:  
В каждой системе управления роботом, с которой есть сетевое соединение, можно выбрать любой проект и передать в WorkVisual. Это возможно и в том случае, если этот проект еще не существует на данном ПК.



- Сравнение проекта с другим проектом и принятие различий при необходимости:  
 Проект можно сравнить с другим проектом. Это может быть проект в системе управления роботом или локально сохраненный проект. Пользователь может решить для каждого отличия по отдельности, сохранить ли состояние как в текущем проекте или принять состояние из другого проекта.
- Передача проектов в систему управления роботом.
- Структурирование и соединение магистральных шин.
- Редактирование конфигурации системы безопасности.
- Автономное программирование роботов.
- Управление длинными текстами.
- Функция диагностики.
- Просмотр системных данных управления роботом онлайн.
- Конфигурирование трассировки, начало записи, анализ трассировки (с помощью осциллоскопа).

**Структура и функционирование панели управления WorkVisual**

По умолчанию в панели управления видны не все элементы, их можно показывать и скрывать по необходимости.

Кроме показанных здесь окон и редакторов, имеются другие. Их можно открыть с помощью команд меню **Окно** и **Редакторы**.

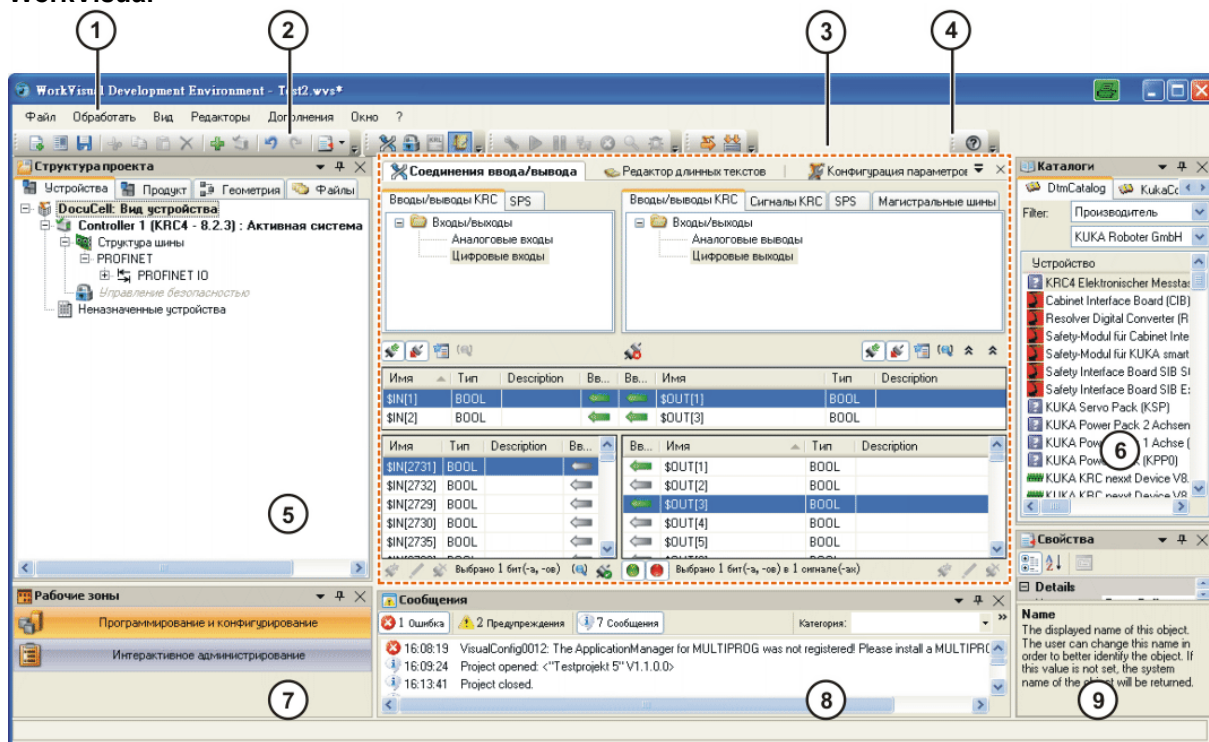


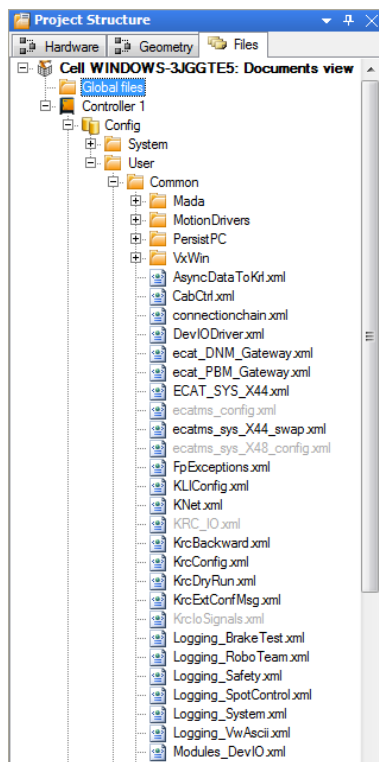
Рис. 18-14: Обзор интерфейса пользователя

Поз.	Описание
1	Строка меню
2	Строки кнопок
3	Область редакторов  Если открыт один из редакторов, он отображается здесь. Одновременно можно открыть несколько редакторов, как показано в примере. В этом случае они располагаются одно под другим, и их можно выбирать с помощью вкладок.
4	Кнопка справки

Поз.	Описание
5	Окно Структура проекта
6	Окно Каталоги В этом окне отображаются все добавленные каталоги. Элементы в каталогах можно добавлять в окне во вкладках <b>Устройства</b> или <b>Геометрия</b> путем перетаскивания.
7	Окно <b>Рабочие зоны</b>
8	Окно <b>Сообщения</b>
9	Окно <b>Свойства</b> При выборе объекта его свойства отображаются в этом окне. Свойства можно изменить. Некоторые свойства в серых полях изменить нельзя.

## Окно Структура проекта

### ■ Вкладка «Структура проекта»



**Рис. 18-15:** Пример: автоматически сгенерированные файлы выделены серым

### ■ Устройства:

На вкладке **Устройства** показана взаимосвязь устройств. Здесь можно присваивать отдельные устройства системе управления роботом.

### ■ Геометрия:

На вкладке **Геометрия** выводятся технические данные для внешних кинематик – например, дополнительных осей.

### ■ Файлы:

На вкладке **Файлы** содержатся все программные файлы и файлы конфигурации, которые относятся к данному проекту.

Цветовое отображение имен файлов:

- автоматически созданные файлы (функцией **Создать код**): серый;
- файлы, добавленные в WorkVisual вручную: синий;

- файлы, переданные в WorkVisual системой управления роботом: черный.

### Проводник проектов

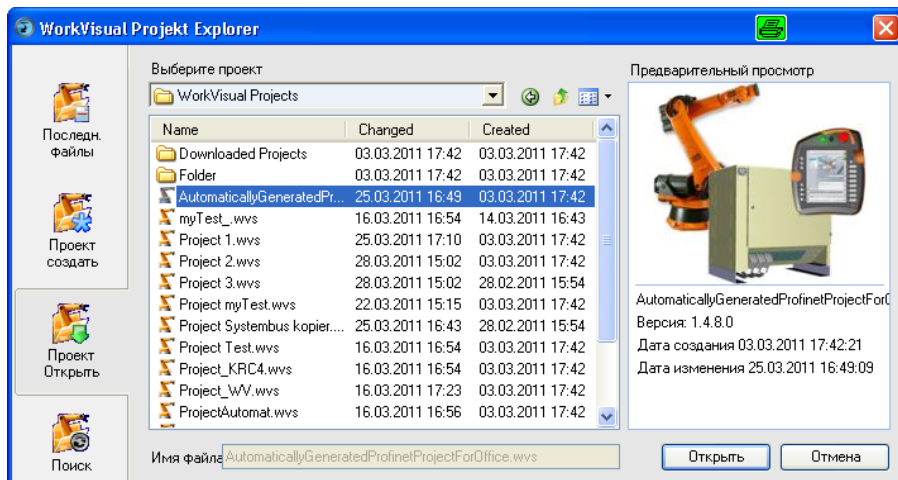


Рис. 18-16: Проводник проектов

- **Последние файлы** показывает последние использованные файлы
- **Создать проект** генерирует
  - новый пустой проект
  - новый проект с помощью макета
  - новый проект на базе существующего проекта
- **Открыть проект** используется для открытия существующего проекта
- **Поиск** используется, чтобы можно было загрузить проект из системы управления роботом

### Указание

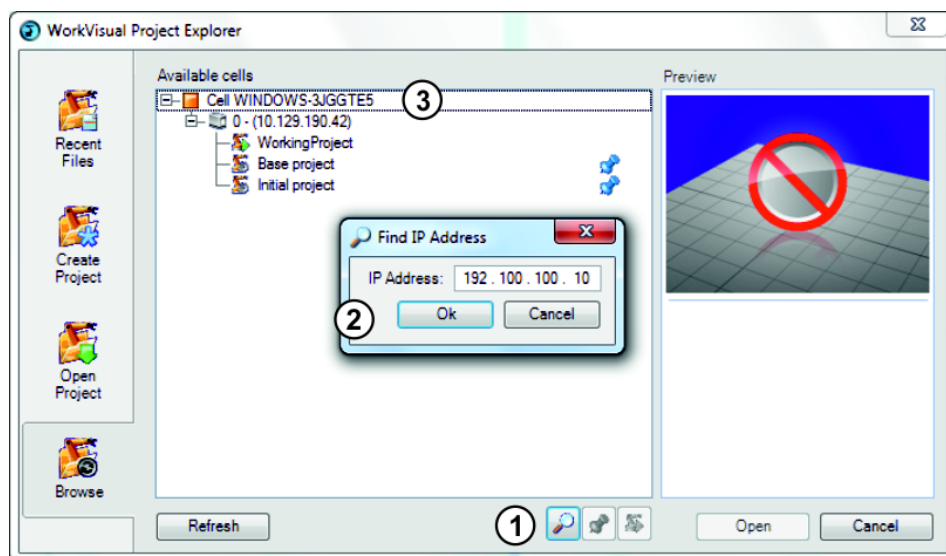


Рис. 18-17: Find IP Address

При определенных обстоятельствах может случиться, что несмотря на существующее сетевое соединение в окне «Поиск» будут отсутствовать системы управления. Устранить эту ситуацию можно следующим образом:

- В проводнике WorkVisual нажать на кнопку **Луна** (1).

- Откроется окно **Find IP Address** (2).
- Ввести нужный IP-адрес шкафа управления.
- Принять IP-адрес нажатием **ОК** (2).
- Система управления появится в окне **Доступные ячейки** (3).

#### Порядок действий при загрузке проекта с помощью WorkVisual

В каждой системе управления роботом, с которой есть сетевое соединение, можно выбрать проект и передать в WorkVisual.

Это возможно и в том случае, если этот проект еще не существует на данном ПК.

Проект сохраняется в каталоге **Собственные файлы\Проекты WorkVisual\Загруженные проекты**.

1. Выберите последовательность меню **Файл > Просмотреть проект**. Откроется **Проводник проектов**. Слева выбрана вкладка **Поиск**.
2. В области **Доступные ячейки** раскройте узел нужной ячейки. Отображаются все системы управления роботами этой ячейки.
3. Раскройте узел нужной системы управления роботом. Отображаются все проекты.
4. Выделите нужный проект и нажмите кнопку **Открыть**. Проект открывается в WorkVisual.

### 18.3.2 Сравнение проектов с помощью WorkVisual

#### Описание

- Один проект в WorkVisual можно сравнить с другим проектом
- Это может быть проект в системе управления роботом или локально сохраненный проект
- Различия наглядно перечисляются, может быть отображена детальная информация
- Пользователь может решить для каждого отличия по отдельности,
  - сохранить ли состояние как в текущем проекте или принять состояние из другого проекта
  - или принять состояние из другого проекта

#### Принцип сравнения проектов

Сведение проекта

- Окно выбора **Сравнить проект**

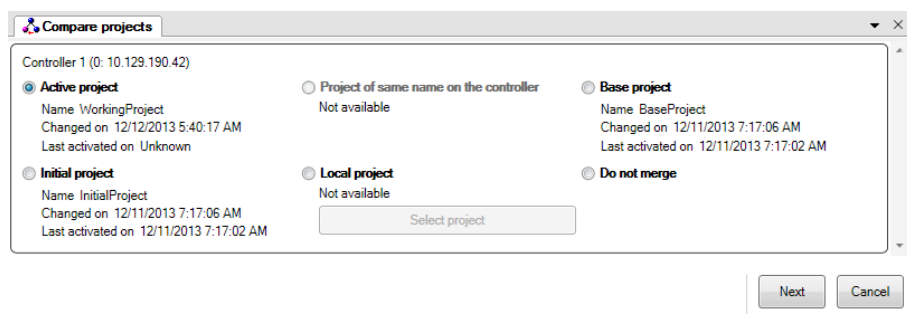
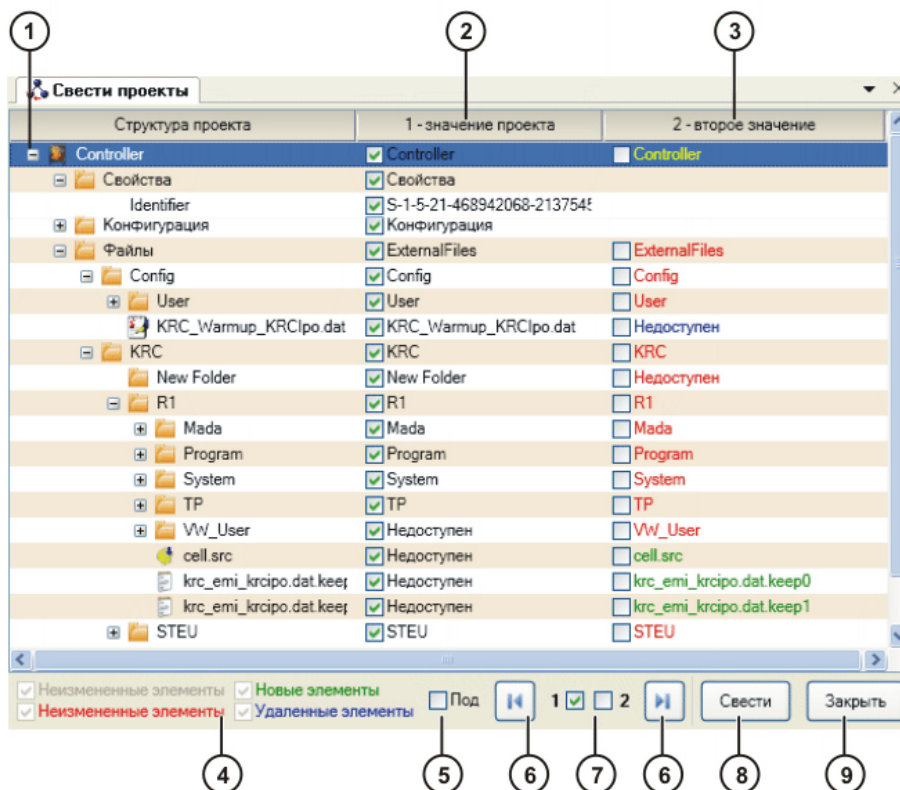


Рис. 18-18: Выбор проекта для сравнения

- Активный проект
- Одноименный проект в системе управления (возможно только с сетевым соединением)
- Базовый проект
- Исходный проект
- Локальный проект (с ноутбука)

**Сравнение:**

Будет отображена сводка различий между проектами. Для каждого отличия можно выбрать, какое состояние следует принять.



**Рис. 18-19: Пример: обзор различий**

Поз.	Описание
1	<p>Узел системы управления роботом. Различные области проекта отображаются в подузлах. Для просмотра сравнений открыть узел.</p> <p>Если имеются несколько систем управления роботом, они перечислены одна под другой.</p> <ul style="list-style-type: none"> <li>В строке галочку следует установить у того значения, которое должно быть передано. (Альтернативно: использовать окошки меток в нижней строке.)</li> <li>Флажок у <b>Недоступен</b> означает, что элемент не записан или, если он уже имелся, удален из проекта.</li> <li>Если поставить флажок у узла, флажки будут автоматически поставлены рядом со всеми подчиненными элементами. Если снять флажок у узла, флажки будут автоматически сняты и рядом со всеми подчиненными элементами. Однако подчиненные элементы можно отмечать и по отдельности.</li> <li>Закрашенное окошко означает следующее: выбран, по меньшей мере, один из подчиненных элементов, но не все.</li> </ul>
2	Состояние в проекте, открытом в WorkVisual.
3	Состояние в сравниваемом проекте.
4	Фильтр для активации и скрытия разных типов различий.
5	TRUE: для отмеченных строк в сводке отображаются подробные сведения.
6	<p>Стрелка «назад»: фокус окна переходит к предыдущему различию.</p> <p>Стрелка «вперед»: фокус окна переходит к следующему различию.</p> <p>При этом закрытые узлы открываются автоматически.</p>

Поз.	Описание
7	Флажки показывают состояние строки, на которой находится фокус. Флажки можно ставить и снимать не только непосредственно в строке, но и здесь.
8	Передача выбранных изменений в открытый проект.
9	Заккрытие окна <b>Свести проекты</b> .

#### Описание цветов:

Столбец	Описание
<b>Структура проекта</b>	Цвет отображения каждого элемента соответствует столбцу, в котором был выбран этот элемент.
<b>WorkVisual</b>	Все элементы отображаются черным цветом.
<b>Выбранное значение</b>	<ul style="list-style-type: none"> <li>■ Зеленый: элементы, которые не присутствуют в открытом проекте, но есть в сравниваемом проекте.</li> <li>■ Синий: элементы, которые присутствуют в открытом проекте, но отсутствуют в сравниваемом проекте.</li> <li>■ Красный: все остальные элементы. К ним относятся также вышестоящие элементы, которые содержат подэлементы разных цветов.</li> </ul>

#### Порядок действий при сравнении проектов

1. В WorkVisual выберите последовательность меню **Дополнения > Сравнение проектов**. Открывается окно **Сравнение проектов**.

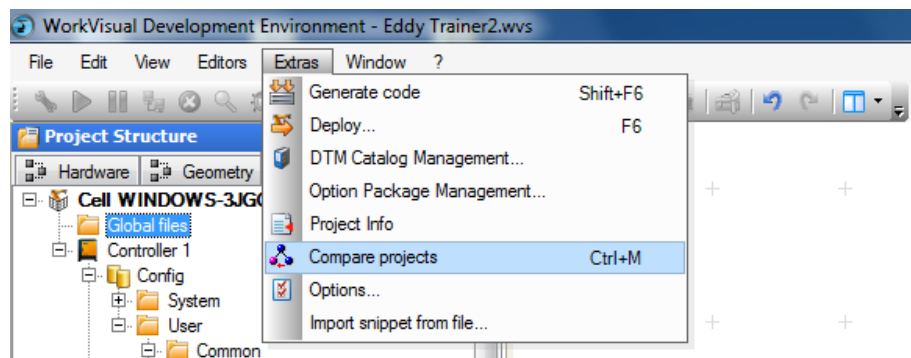


Рис. 18-20: Сравнение проектов

2. Выберите проект, с которым следует сравнить текущий проект WorkVisual, например одноименный проект из реальной системы управления роботом.

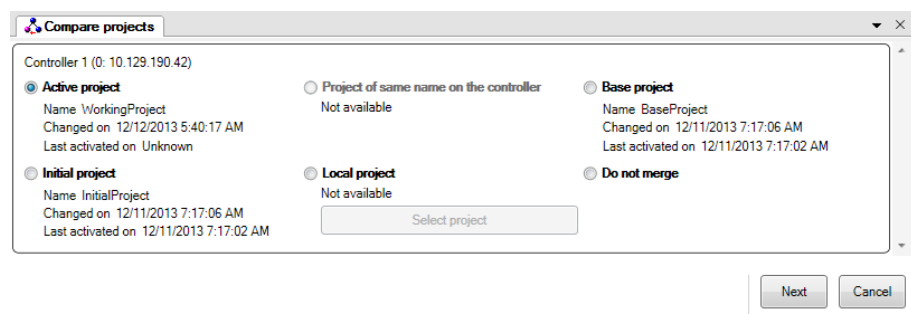


Рис. 18-21: Выбор проекта для сравнения

3. Нажмите кнопку **Продолжить**. Появится индикатор выполнения. Если проект содержит несколько систем управления роботами, для каждой будет отображен отдельный индикатор.



4. Когда выполнение будет завершено и будет отображаться состояние: **Готов к выполнению сведения**: Нажмите кнопку **Показать отличия**. Будет отображена сводка различий между проектами.  
Если различия не обнаружены, появляется соответствующее сообщение. Продолжите с шага 8. После этого дальнейшие действия не нужны.
5. Для каждого различия выберите, какое состояние следует принять. Не следует делать этого для всех различий сразу.  
Если этот вариант подойдет, можно оставить все по умолчанию.
6. Нажмите **Свести**, чтобы передать изменения в WorkVisual.
7. Шаги 5 и 6 можно повторять любое количество раз. Таким образом, разные области можно редактировать снова и снова.  
Если больше различий нет, появляется соответствующее сообщение. **Другие отличия отсутствуют**.
8. Закройте окно **Сравнение проектов**.
9. Если в системе управления роботом в проекте были изменены параметры дополнительных осей, их нужно обновить в WorkVisual:
  - Для этой дополнительной оси следует открыть окно **Конфигурация параметров машины**.
  - В области **Общие параметры машины, зависящие от осей** нажмите кнопку импорта параметров машины.  
Данные обновляются.
10. Сохраните проект.

Пример сравнения проектов

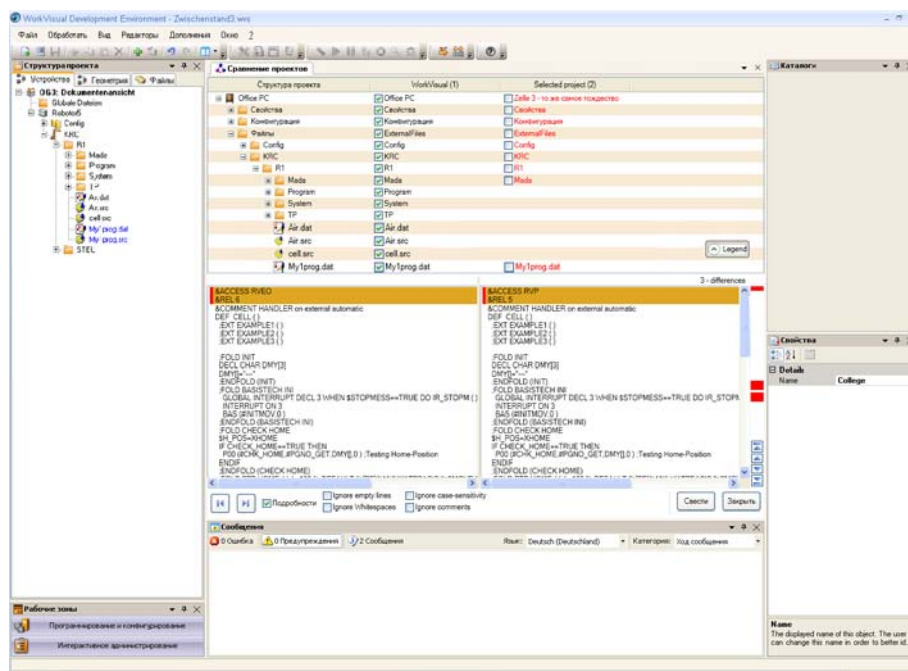


Рис. 18-22: Пример: Сравнение проектов Обзор

Решение, какое состояние файла (файлов) следует принять.

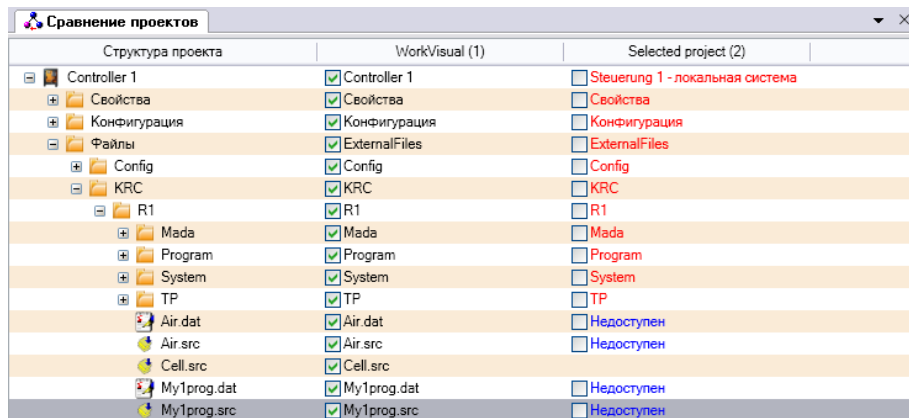


Рис. 18-23: Пример: Свести проект

При активации изображения **Детали** показываются различия между файлами

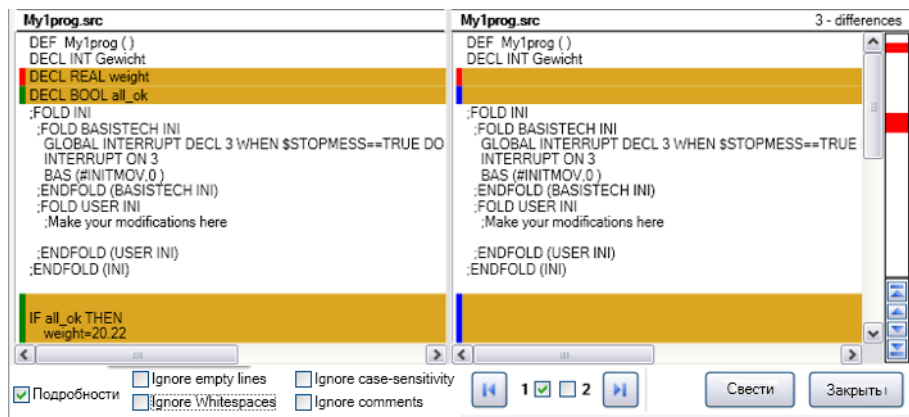


Рис. 18-24: Пример: Изображение Детали активировано

### 18.3.3 Передача проекта в систему управления роботом (установка)

#### Описание

- После изменений в проекте его необходимо передать из WorkVisual в систему управления.
- Этот процесс называется у КУКА «Установить».
- При передаче проекта в систему управления роботом сначала всегда создается код.
- Сетевое соединение с реальной системой управления роботом – это условие для выполнения процесса «Установить».



Если в реальной системе управления роботом имеется проект, который ранее был передан, но еще не активирован, при передаче нового проекта он стирается.


При передаче и активации проекта существующий в реальной системе управления роботом одноименный проект также стирается (после предупредительного запроса).

#### Функции

##### Создать код

- Таким образом можно отдельно создать код, чтобы предварительно убедиться в том, что он создается без ошибок.
- Вызов осуществляется посредством
  - последовательности меню **Дополнения > Создать код**.



- или кнопкой 
- Код отображается в окне **Структура проекта** на вкладке **Файлы**. Автоматически сгенерированный код отображается светло-серым цветом.

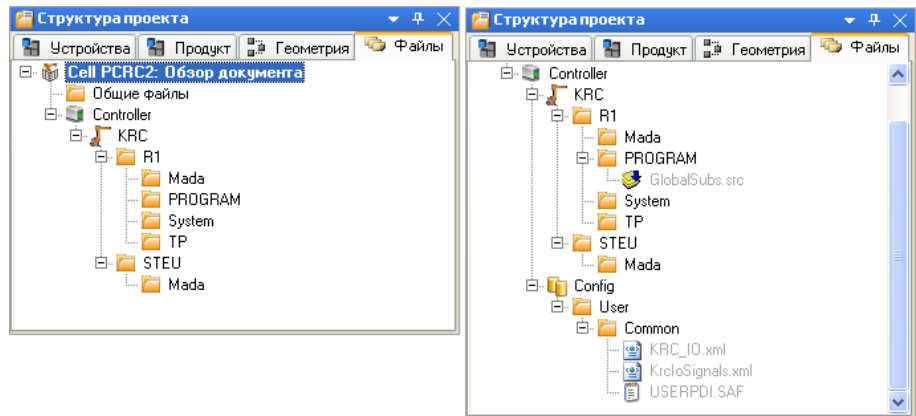



Рис. 18-25: Пример создания кода: прежде - после

- Создается код. После завершения этого процесса в окне сообщений показываются следующие сообщения: **Проект <"{0}" V{1}> был скомпилирован. Результаты приведены в дереве файлов.**

**Указание**

1. В строке меню нажать кнопку **Установить ...** . Откроется окно **Передача проекта**.

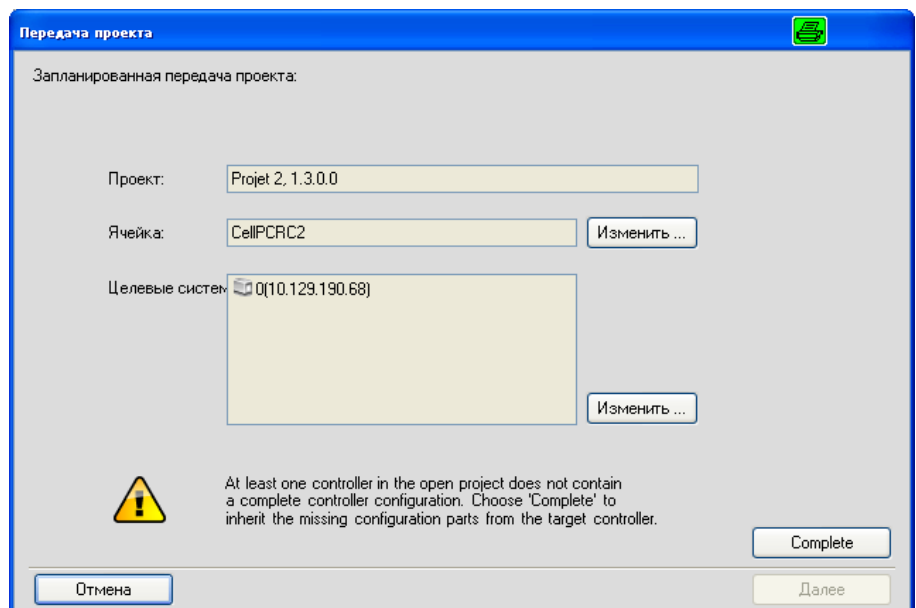
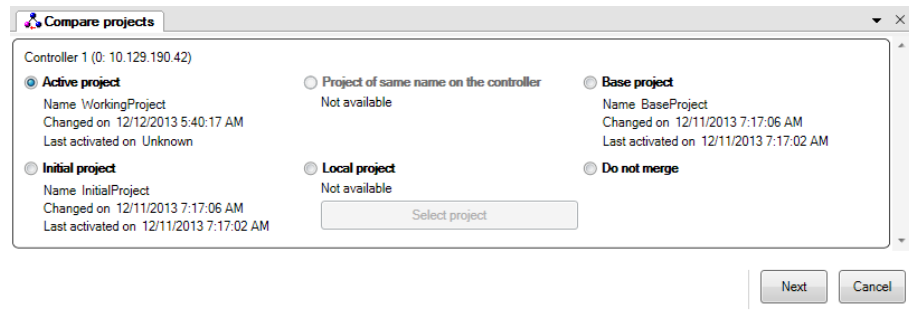



Рис. 18-26: Обзор с указанием на неполную конфигурацию

2. Если передаваемый проект еще не был возвращен из системы управления роботом в WorkVisual, он содержит не все данные конфигурации (к файлам конфигурации относятся файлы параметров машины, файлы конфигурации системы безопасности и многие другие). Сообщение об этом выводится на экран.
  - Если это сообщение не отображается: продолжить с шага 13.
  - Если это сообщение отображается: продолжить с шага 3.

3. Нажать кнопку **Дополнить**. Будет отображен следующий предупредительный запрос: **Проект следует сохранить, и активная система управления будет сброшена. Продолжить?**
4. Нажать в окне запроса кнопку **Да**. Откроется окно **Свести проекты**.



**Рис. 18-27: Выбор проекта для сравнения**

5. Выбрать проект, данные конфигурации которого требуется передать, например, активный проект из реальной системы управления роботом.
6. Нажать кнопку **Далее**. Появится индикатор выполнения. Если проект содержит несколько систем управления роботами, для каждой будет отображен отдельный индикатор.
7. Когда выполнение будет завершено и будет отображаться состояние, а также **Готов к выполнению сведения**: Нажать кнопку **Показать отличия**.  
Будет отображена сводка различий между проектами.
8. Для каждого различия выбрать, какое состояние следует принять. Не следует делать этого для всех различий сразу.  
При необходимости можно оставить выбор по умолчанию.
9. Нажать **Свести**, чтобы передать изменения.
10. Шаги 8 и 9 можно повторять любое количество раз. Таким образом, разные области можно редактировать снова и снова.  
Если больше различий нет, появляется соответствующее сообщение. **Другие отличия отсутствуют**.
11. Закрыть окно **Сравнение проектов**.
12. В строке меню нажать кнопку **Установить...** . Будет снова отображена сводка присвоения ячеек. Сообщение о неполной конфигурации больше не отображается.

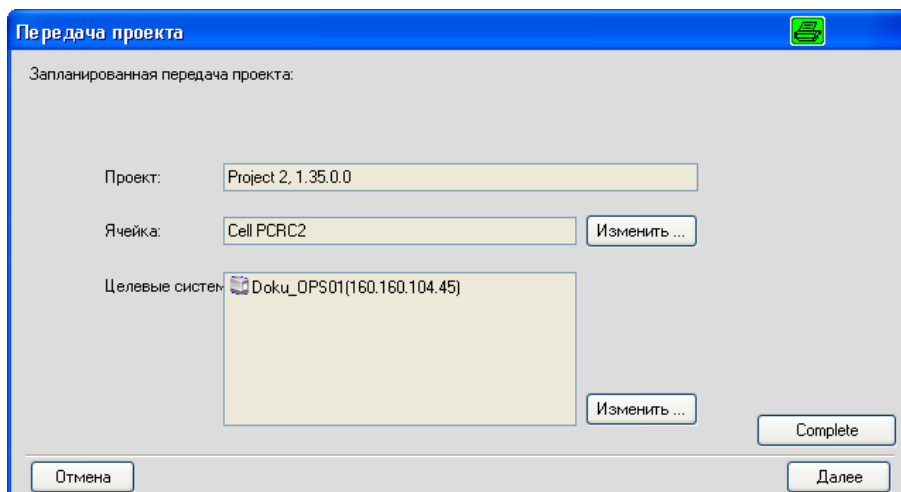


Рис. 18-28: Обзор

13. Нажать кнопку **Далее**. Начинается создание программы. Когда индикатор выполнения покажет 100 %, программа создана, а проект передан.
14. Нажать кнопку **Активировать**.

**⚠ ОСТОРОЖНО** В режимах AUT и AUT EXT проект активируется без предварительного запроса, если выполнены только изменения программы.

15. Только в режимах работы T1 и T2: KUKA smartHM1 задает вопрос *Допустить активацию проекта [...]?*. Дополнительно отображается, будет ли при активации стерт проект, и если да, то какой.  
Если никакой нужный проект не стирается, подтвердить запрос в течение 30 минут, нажав кнопку **Да**.
16. Отобразится сводка изменений по сравнению с еще активным проектом в системе управления роботом. Установив флажок **Подробности**, можно отобразить подробности изменений.
17. В окне сводки отобразится предупредительный запрос *Продолжить?*. Нажать кнопку **Да**. Система управления роботом активирует проект. В WorkVisual отображается подтверждение.

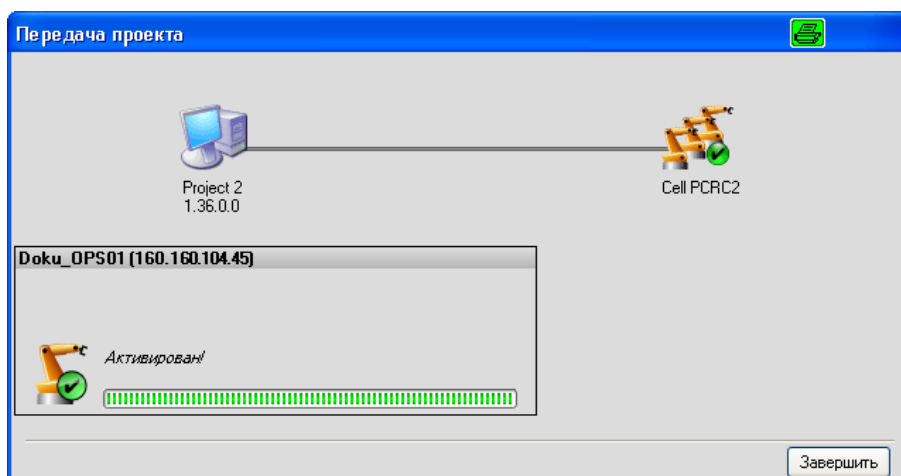


Рис. 18-29: Подтверждение в WorkVisual

18. Закрыть окно **Передача проекта**, нажав кнопку **Завершить**.
19. Если не ответить на вопрос системы управления роботом в течение 30 минут, проект все же будет передан. Однако он не будет активиро-

ван в системе управления роботом. В этом случае проект можно активировать отдельно.

**⚠ ОСТОРОЖНО** После активации проекта в системе управления роботом там следует проверить конфигурацию системы безопасности! В противном случае робот может работать с неверными данными. Это может привести к летальному исходу, тяжелым травмам или значительному материальному ущербу.

**⚠ ОСТОРОЖНО** Если активировать проект не удалось, в WorkVisual выдается сообщение об ошибке. В этом случае нужно выполнить одно из следующих действий:

- Заново активировать проект. (Тот же или новый).
- Или заново провести холодный пуск системы управления роботом.

#### 18.3.4 Активация проекта в системе управления роботом

**Описание**

- Проект можно активировать прямо в системе управления роботом.
- Проект также можно активировать из WorkVisual в системе управления роботом  
(дальнейшие пояснения здесь отсутствуют, см. онлайн-документацию по WorkVisual).

**Функция управления проектами**

##### Общая информация

- Система управления роботом имеет возможность управлять несколькими проектами в системе управления.
- Все функции доступны только в группе пользователя **Эксперт**
- Для вызова выберите:
  - последовательность меню **Файл > Управление проектами**
  - на панели управления клавишу **Символ WorkVisual** и затем кнопку **Открыть**

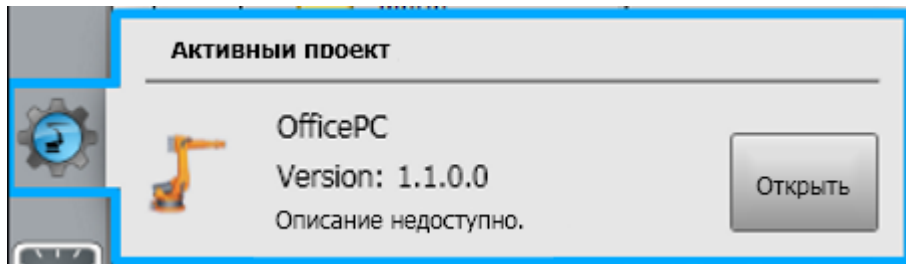


Рис. 18-30: Индикация проекта на пользовательском интерфейсе

Обращение/управление

■

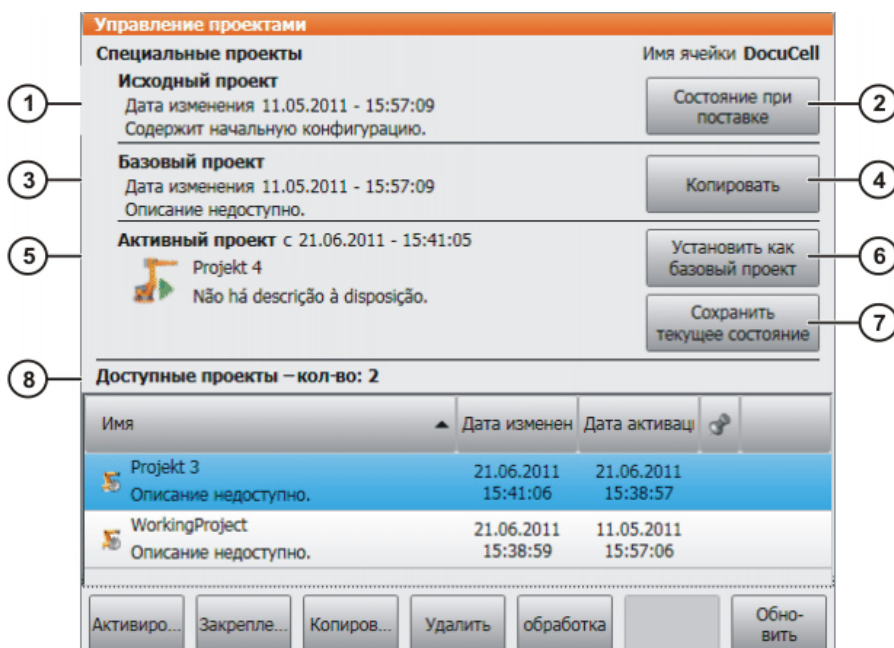


Рис. 18-31: Окно управления проектом

Поз.	Описание
1	Отображение исходного проекта.
2	Восстановление исходного состояния системы управления роботом. Доступно только для группы пользователей «Эксперт» и выше.
3	Отображение базового проекта.
4	Создание копии базового проекта. Доступно только для группы пользователей «Эксперт» и выше.
5	Отображение активного проекта.
6	Сохранение активного проекта как базового. Активный проект остается активным. Доступно только для группы пользователей «Эксперт» и выше.
7	Создание закрепленной копии активного проекта. Доступно только для группы пользователей «Эксперт» и выше.
8	Список проектов. Активный проект не отображается.

- Кроме обычных проектов, окно **Управление проектами** содержит следующие специальные проекты:


Проект	Описание
Исходный проект	Исходный проект существует всегда. Пользователь не может изменить его. Он содержит состояние системы управления роботом на момент поставки.
Базовый проект	<p>Пользователь может сохранить активный проект как базовый. Эта возможность обычно применяется, когда требуется сохранить работоспособное, проверенное состояние проекта.</p> <p>Базовый проект нельзя активировать, а можно только копировать. Пользователь больше не может изменить базовый проект. Однако путем сохранения можно заменить один базовый проект другим (после предупредительного запроса).</p> <p>При активации проекта, который содержит не все файлы конфигурации, недостающие сведения извлекаются из базового проекта. Это может быть, например, при активации проекта из более ранней версии WorkVisual. К файлам конфигурации относятся файлы параметров машины, файлы конфигурации системы безопасности и многие другие.</p>

■ Описание клавиш


Клавиша	Описание
<b>Активировать</b>	<p>Активация выделенного проекта.</p> <p>Если выделенный проект закреплен: создание копии выделенного базового проекта. (Можно активировать только копию закрепленного проекта, но не сам проект.) Затем пользователь может решить, активировать ли копию или активным должен оставаться прежний проект.</p> <p>Доступно только для группы пользователей «Эксперт» и выше.</p>
<b>Закрепление</b>	<p>Закрепленные проекты нельзя изменять, активировать или удалять. Однако их можно копировать или откреплять. Проект можно закрепить, чтобы предотвратить его случайное удаление.</p> <p>Доступно, только если выделен незакрепленный проект. Доступно только для группы пользователей «Эксперт» и выше.</p>
<b>Отстыковка</b>	<p>Отмена закрепления проекта.</p> <p>Доступно, если выделен закрепленный проект. Доступно только для группы пользователей «Эксперт» и выше.</p>
<b>Копирование</b>	<p>Копирование выделенного проекта.</p> <p>Доступно только для группы пользователей «Эксперт» и выше.</p>
<b>Удалить</b>	<p>Удаление выделенного проекта.</p> <p>Доступно, если выделен неактивный и не закрепленный проект. Доступно только для группы пользователей «Эксперт» и выше.</p>

Клавиша	Описание
<b>Редактировать</b>	Открывает окно, в котором можно изменить имя и/или описание выделенного проекта.  Доступно, только если выделен незакрепленный проект. Доступно только для группы пользователей «Эксперт» и выше.
<b>Обновить</b>	Обновление списка проектов. Таким способом можно просмотреть проекты, которые были переданы в систему управления роботом после открытия этого окна.

**Порядок действий**

 Ограничение: если активация может вызвать изменения конфигурации системы безопасности в области **Параметры связи**, следует выбрать группу пользователей «Специалист по обслуживанию системы безопасности» или выше.  
Если выбран режим работы AUT или AUT EXT: проект можно активировать только в том случае, если это ведет только к изменению программ KRL. Если проект содержит настройки, которые могут привести к другим изменениям, активировать его невозможно.

1. Выберите последовательность меню **Файл > Управление проектами**. Откроется окно **Управление проектами**.
2. Отметьте нужный проект и нажмите **Активировать**.
3. KUKA smartHMI задает вопрос *Вы хотите разрешить активацию проекта [...]?*. Дополнительно отображается, будет ли при активации стерт проект, и если да, то какой.  
Если никакой нужный проект не стирается, подтвердите запрос в течение 30 мин, нажав кнопку **Да**.
4. Отобразится сводка изменений по сравнению с еще активным проектом в системе управления роботом. Установив флажок **Подробности**, можно отобразить подробности изменений.
5. В окне сводки отобразится предупредительный запрос *Продолжить?*. Нажмите кнопку **Да**. Система управления роботом активирует проект.

 **ОСТОРОЖНО** После активации проекта в системе управления роботом там следует проверить конфигурацию системы безопасности! В противном случае робот может работать с неверными данными. Это может привести к летальному исходу, тяжелым травмам или значительному материальному ущербу.

**18.4 Редактирование программ KRL с помощью WorkVisual**

- Работа с файлами  
(>>> 18.4.1 "Работа с файлами" Страница 266)
- Работа с редактором KRL  
(>>> 18.4.2 "Работа с редактором KRL" Страница 272)

**18.4.1 Работа с файлами**

- Описание**
- Загрузка имеющегося файла в редактор KRL
  - Добавление нового файла из каталога
  - Добавление внешнего файла

## Принцип использования шаблонов из каталогов

- Перед использованием шаблонов необходимо загрузить соответствующий каталог
- Загрузите через **Файл > Добавить каталог** и активируйте соответствующие шаблоны
  - Шаблоны KRL: *KRL Templates.afc*
  - Шаблоны VW: *VW Templates.afc*
- Каталог шаблонов KRL

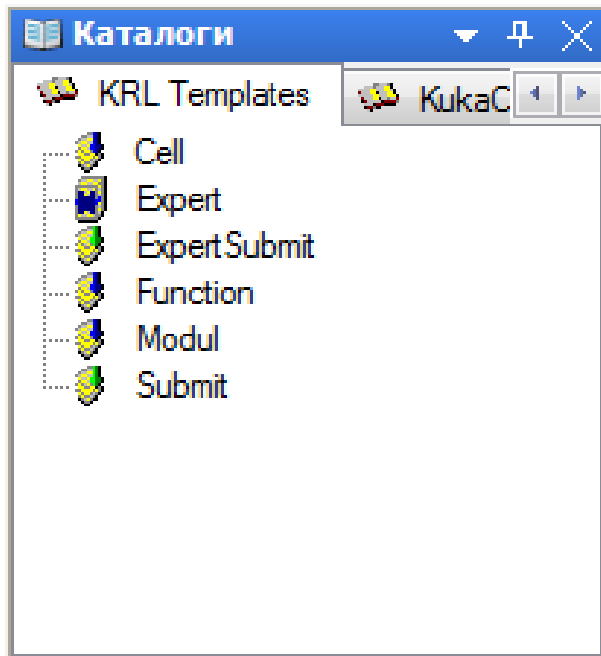


Рис. 18-32: Каталог для шаблонов KRL

## Указания

### Порядок действий при открывании файла (SRC/DAT) в редакторе KRL

1. Перейдите к другому файлу в дереве проекта

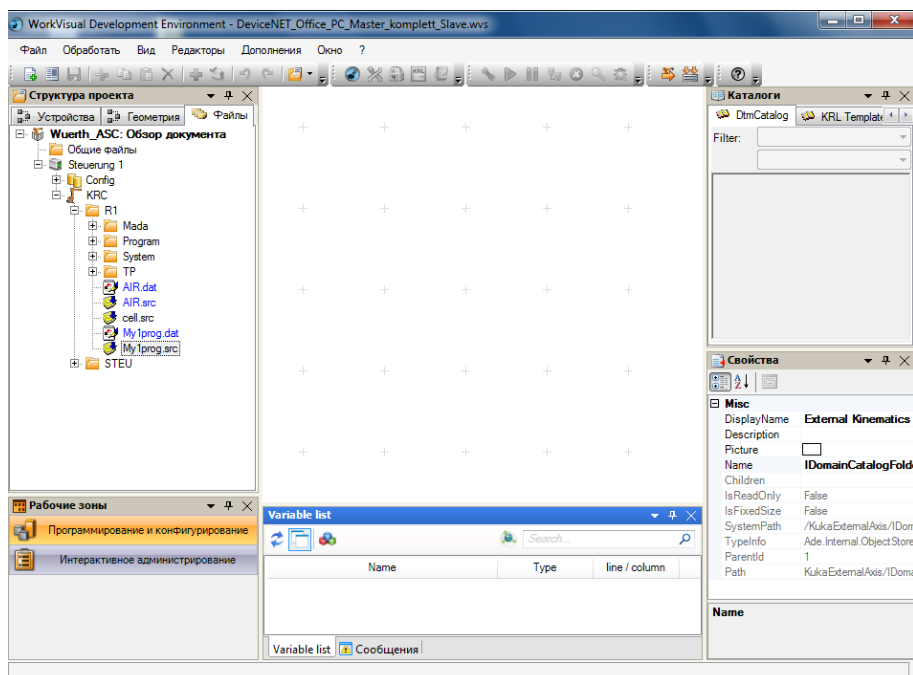


Рис. 18-33: Дерево проекта WorkVisual

2. Разверните директории до R1 включительно



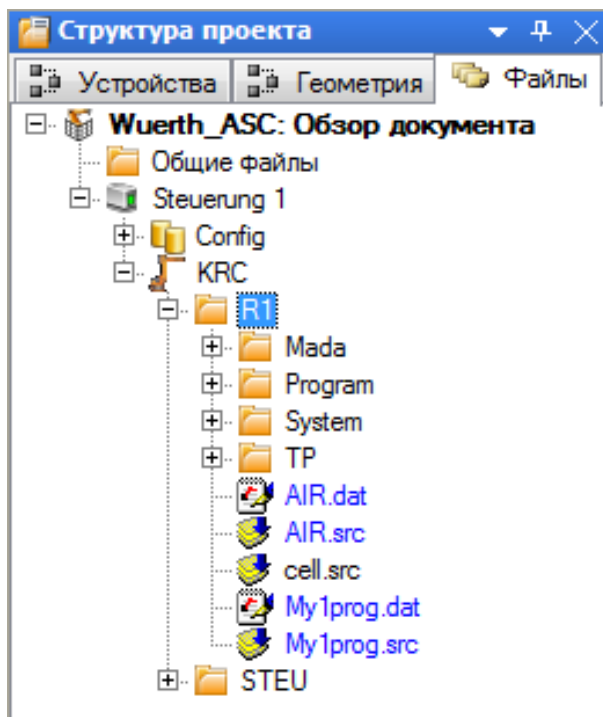


Рис. 18-34: Структура файлов в дереве проекта WorkVisual (R1)

3. Выберите файл и или

- двойным щелчком

- кнопкой в панели инструментов 

- правой кнопкой мыши, через контекстное меню редактора Krl



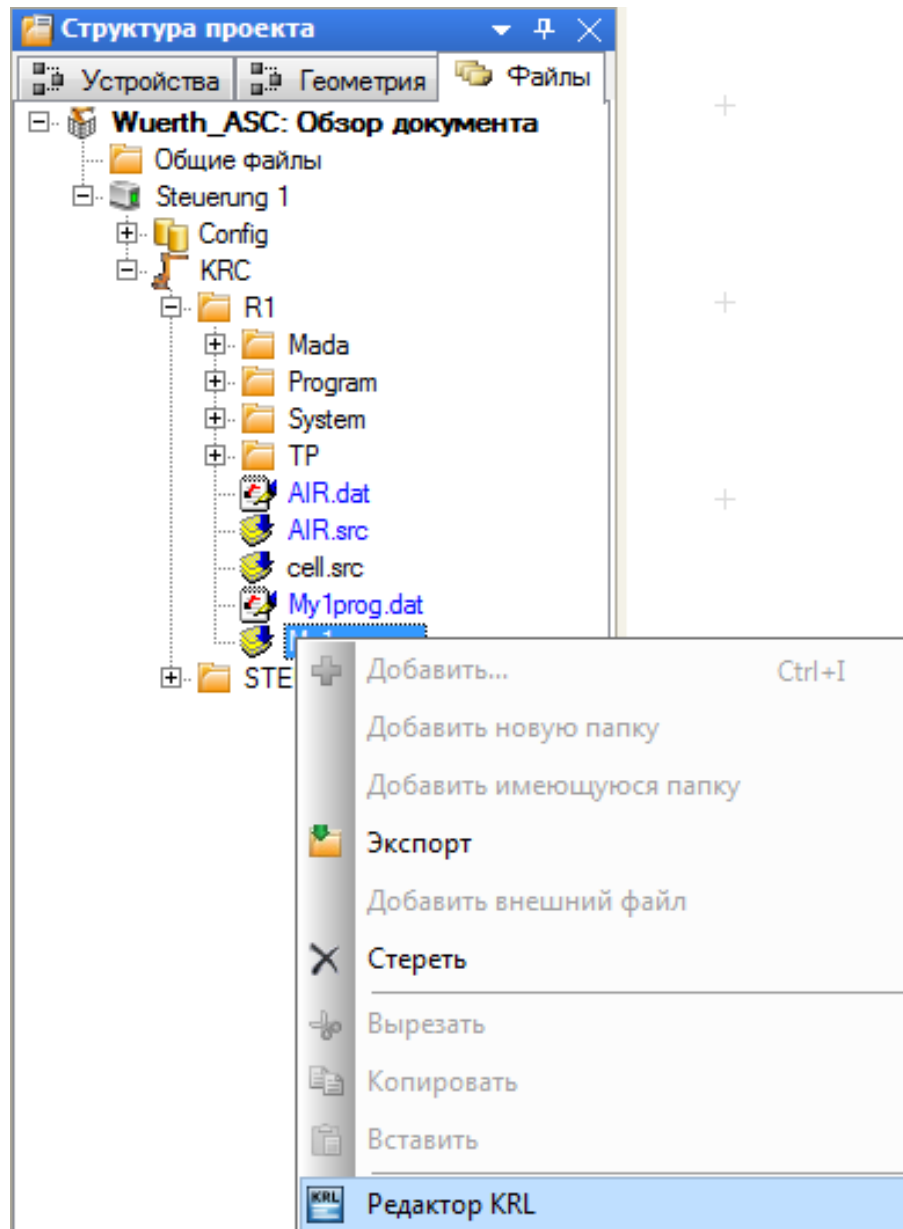


Рис. 18-35: Контекстное меню мыши WorkVisual (редактор KRL)

#### Порядок действий при добавлении файла с помощью шаблонов KRL

1. Перейдите к другому файлу в дереве проекта
2. Разверните директории до R1 включительно
3. Выберите папку для создания нового файла
4. Нажмите правую кнопку мыши и выберите в контекстном меню опцию



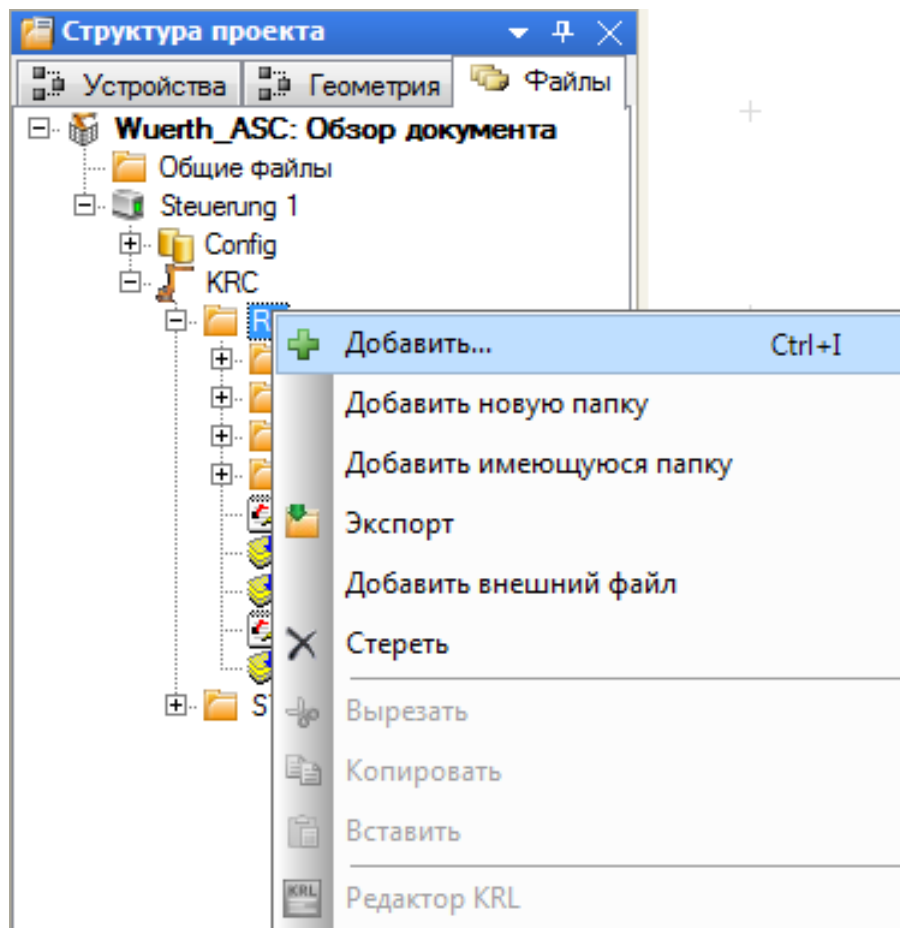


Рис. 18-36: Контекстное меню WorkVisual (Добавить)

5. Выберите шаблон

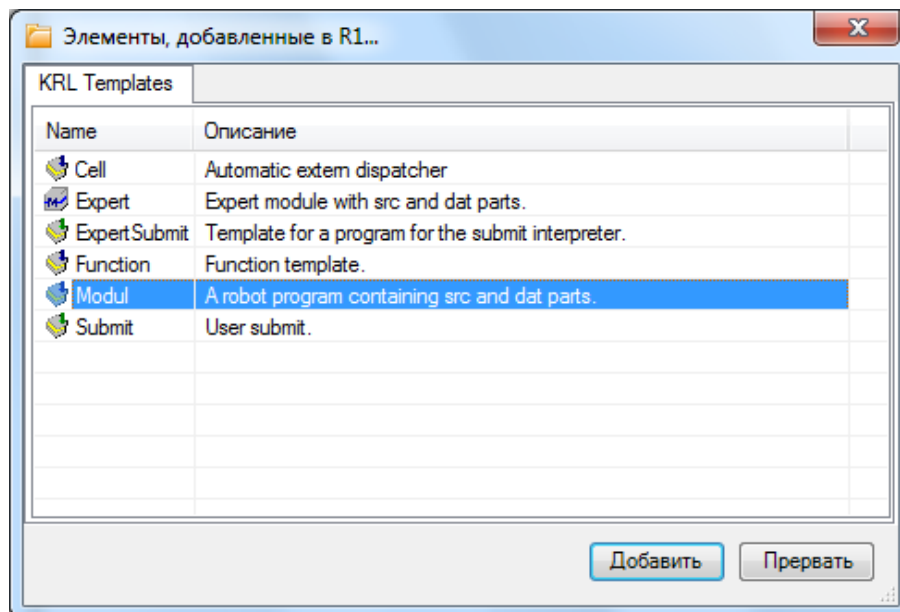


Рис. 18-37: Шаблоны KRL WorkVisual

6. Присвойте имя программе

**Порядок действий при добавлении внешнего файла**

1. Перейдите к другому файлу в дереве проекта
2. Разверните директории до R1 включительно

3. Выберите папку, в которой нужно создать новый файл
4. Нажмите правую кнопку мыши и выберите в контекстном меню опцию «Добавить внешний файл»

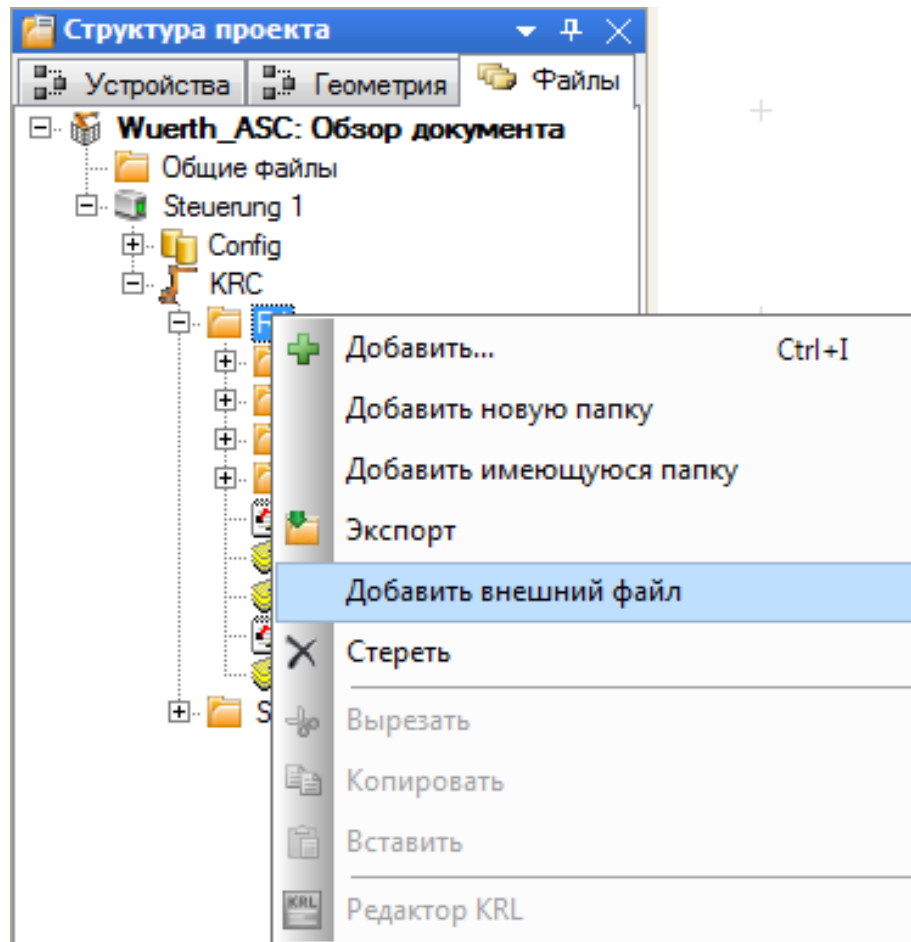


Рис. 18-38: Контекстное меню WorkVisual (Добавить внешний файл)

5. Выберите файл(ы) и нажмите кнопку **Открыть**

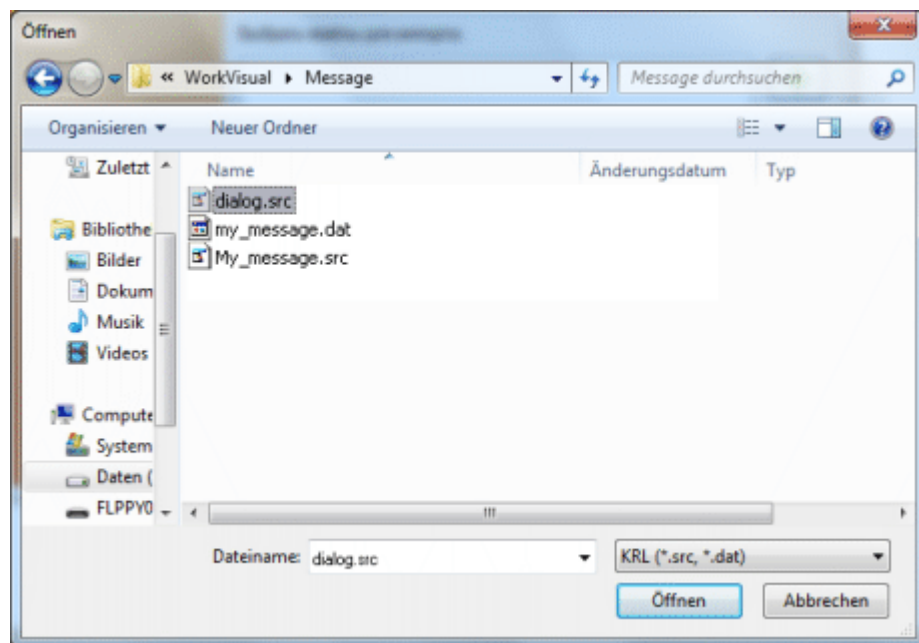


Рис. 18-39: Добавление внешнего файла в WorkVisual

### 18.4.2 Работа с редактором KRL

#### Описание редактора KRL

#### Обработка программы (SRC/DAT)

- посредством прямого ввода KRL
- посредством быстрого ввода команд KRL (фрагментов KRL)
- посредством встроенных формуляров из панели инструментов

#### Свойства редактора KRL

- Конфигурация редактора
- Описание цветов в редакторе KRL
- Распознавание ошибок (анализатор KRL)
- Список переменных
- Дополнительная функция редактирования в редакторе KRL

#### Описание цветов редактора KRL

- Описание цветов  
Редактор KRL распознает компоненты введенного кода и автоматически отображает их в разных цветах.

Компонент кода	Цвет
Ключевые слова KRL (кроме ; FOLD и ; ENDFOLD)	Синий
; FOLD и ; ENDFOLD	Серый
Числа	Темно-синий
Строки (текст в кавычках "...")	Красный
Комментарии	Зеленый
Специальные знаки	Сине-зеленый
Другой код	Черный

- Пример использования цветов

```

15
16 ; FOLD PTP XHOME Vel= 100 % DEFAULT; ; {PE} %MKUKATPBASIS
17 $BWDSTART = FALSE
18 PDAT_ACT=PDEFAULT
19 FDAT_ACT=FHOME
20 BAS (#PTP_PARAMS, 100)
21 $H_POS XHOME
22 PTP XHOME
23 ; ENDFOLD
24
    
```


Рис. 18-40: Пример цветов в редакторе KRL

1	Ключевые слова KRL: синий
2	Комментарий: зеленый
3	FOLD (формуляр): серый
4	Другой код: черный

#### Распознавание ошибок в редакторе KRL

- Редактор KRL имеет функцию автоматического распознавания ошибок
- Распознанные ошибки в программном коде помечаются красным

- Ошибки отображаются в окне сообщений только в том случае, если выбрана категория **Анализатор KRL**.
- Распознаются ошибки KRL и частично структурные ошибки (объявление в неправильном месте программы).
- Опечатки в переменных не распознаются.

 Эта функция распознает не все ошибки. Если ничего не отмечено цветом, это не значит, что в программе нет ошибок.

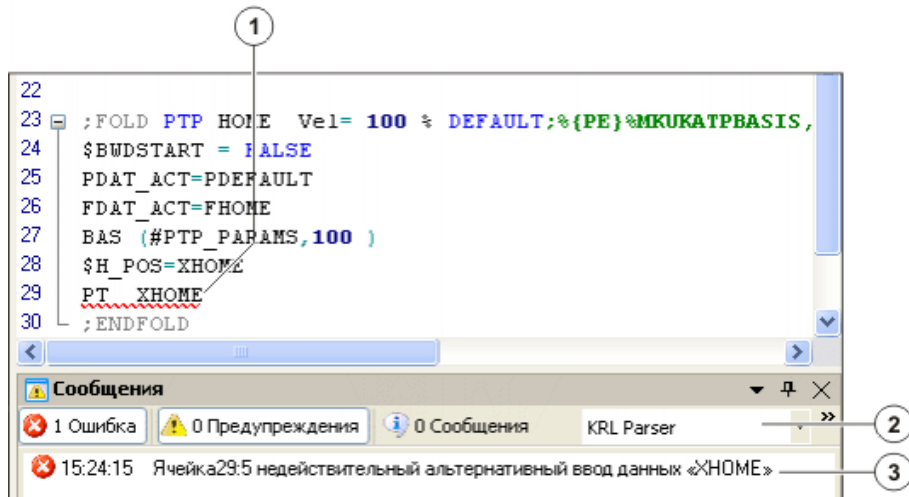


Рис. 18-41: Пример распознавания ошибки

1	Редактор KRL: ошибка выделена красным
2	Окно сообщений: выбрана категория <b>Анализатор KRL</b>
3	Окно сообщений: описание ошибки с номером строки и столбца

### Функция списка переменных

- Все переменные KRL, объявленные в определенном файле, могут быть наглядно отображены в списке.

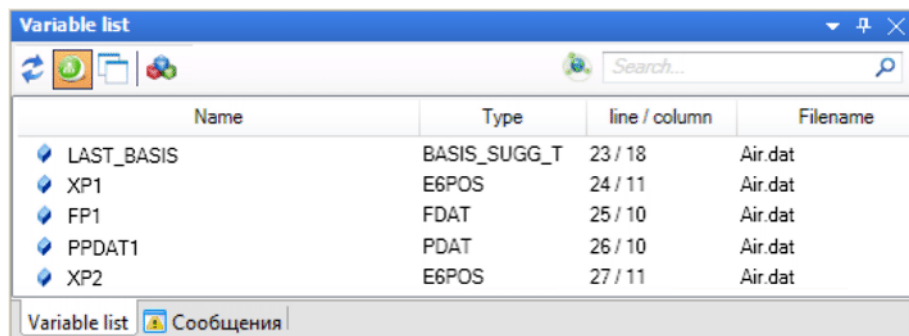


Рис. 18-42: Пример списка переменных

- Для файлов SRC также отображаются переменные из соответствующего файла DAT и наоборот.

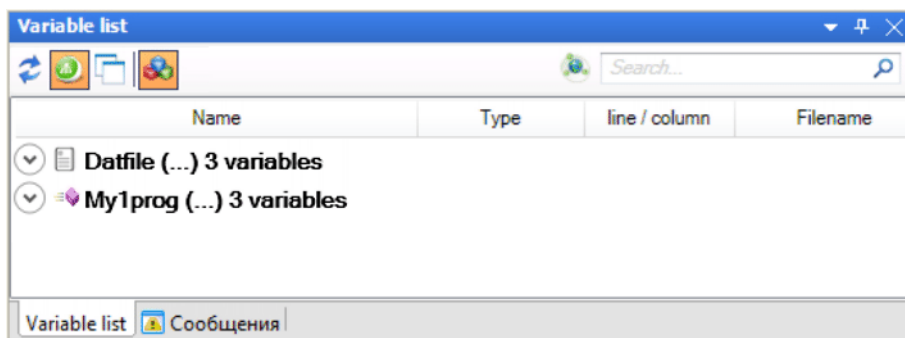


Рис. 18-43: Пример списка переменных (переменные SRC и DAT)

- Введите в поле поиска имя переменной или его часть. Результат поиска будет отображен сразу же.

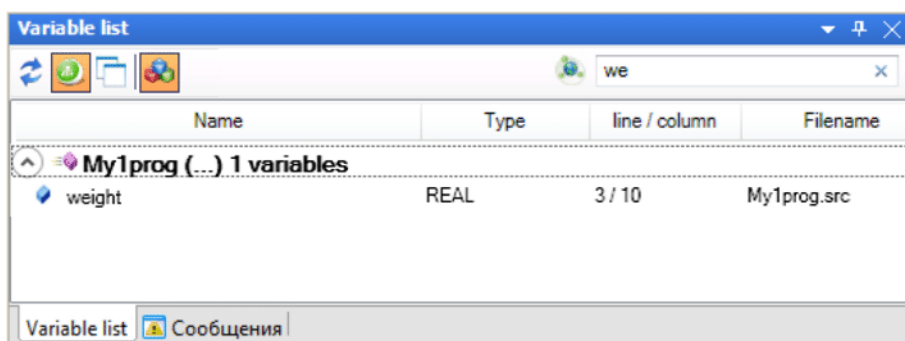


Рис. 18-44: Пример списка переменных (функция поиска)

**i** Уже введенный в поле поиска текст также стирается, если фокус помещен на другой файл. Поэтому для отображения всех переменных файла поле поиска должно быть пусто.

- Возможности настройки

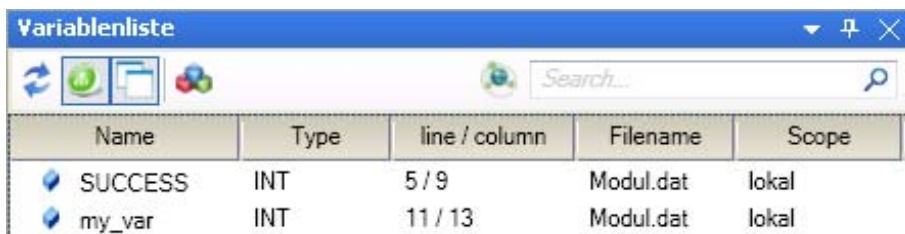







Рис. 18-45: Окно «Список переменных»

Чтобы отсортировать список по конкретному столбцу, наведите указатель на его заголовок и нажмите левую кнопку мыши.

Кнопка	Описание
	<p><b>Считать заново</b></p> <hr/> <p>Все файлы в окне <b>Структура проекта</b> на вкладке <b>Файлы</b> читаются заново. Обновляются глобальные переменные и отношения между файлами.</p>
	<p>Всегда обновляет список, если изменяется выбранный файл в дереве</p> <hr/> <p>При нажатой кнопке: если в окне <b>Структура проекта</b> на вкладке <b>Файлы</b> выделен файл, список переменных обновляется.</p> <p>При ненажатой кнопке: если на вкладке <b>Файлы</b> выделен файл, список переменных не изменяется.</p>
	<p>Обновление списка, если изменяется текущий редактор</p> <hr/> <p>При нажатой кнопке: если в редакторе KRL выбрана вкладка файла, список переменных обновляется.</p> <p>При ненажатой кнопке: если в редакторе KRL выбрана вкладка файла, список переменных не обновляется.</p>
	<p>Группирование переменных по локальным подфункциям (SRC/DAT)</p> <hr/> <p>При нажатой кнопке: индикация сортируется по типам данных. (В рамках этой сортировки можно дополнительно сортировать по столбцам.)</p> <p>При ненажатой кнопке: индикация не сортируется по типам данных.</p>
	<p>При нажатой кнопке: поиск охватывает все глобальные переменные.</p> <hr/> <p>При ненажатой кнопке: поиск охватывает все локальные переменные текущего файла.</p>

### Переименование переменных

Имя переменной с помощью одной операции можно изменять в любом месте, где она встречается.

Это возможно также, если переменная объявлена в файле DAT и используется в нескольких файлах SRC.

Порядок действий

1. В любом месте выделите требуемую переменную.
2. Нажмите правую кнопку мыши и выберите в контекстном меню **Переименовать**.
3. Откроется окно. Измените имя и подтвердите, нажав **ОК**.

### Переход для объявления переменной

Порядок действий

1. Установите курсор на имя переменной или прямо перед первой буквой или сразу после последней.
2. Нажмите правую кнопку мыши и выберите в контекстном меню **Перейти к объявлению**.



**Дополнительные функции редактирования**

Часто используемые функции редактирования можно вызвать в контекстном меню в пункте **Редактирование**. К ним относятся следующие:

- **Вырезать, Вставить, Копировать, Удалить**
- **Отмена, Восстановить**
- **Найти ..., Заменить ...**
- **Перейти к...**
- **Отметить все**

В контекстном меню в разделе **Расширенные** имеются дополнительные функции редактирования:

Расширенные > ...	Описание
<b>Использовать вкладки</b>	Замена пробелов табуляторами в выделенной области.  Необходимое условие: Настройка конфигурации <b>Использовать табуляторы</b> активна.
<b>Удалить вкладки</b>	Замена табуляторов пробелами в выделенной области.
<b>Увеличить отступ</b>	Вставка пробела или табулятора на месте курсора.
<b>Уменьшить отступ</b>	Удаление табулятора или пробела слева от курсора.
<b>Преобразовать в комментарий</b>	Вставка точки с запятой в начале выделенной строки.
<b>Преобразовать из комментария</b>	Удаление точки с запятой в начале выделенной строки.
<b>Закрыть все формуляры</b>	Закрывает все формуляры открытого файла.
<b>Открыть все формуляры</b>	Открывает все формуляры открытого файла.

**Указания**

**Быстрый ввод команд KRL (фрагментов KRL)**

Необходимо запрограммировать цикл FOR. Чтобы не пришлось вводить весь синтаксис `FOR ... = ... TO ... STEP ...`, используются фрагменты кода KRL. Вручную останется только заполнить варьируемые места в синтаксисе.

**Порядок действий при программировании с использованием фрагментов кода**

1. Установите курсор в нужное место.
  - Нажмите правую кнопку мыши и выберите в контекстном меню пункт **Добавить фрагмент кода**. Активируется поле списка. Наведите указатель на нужную инструкцию и дважды нажмите левую кнопку мыши.

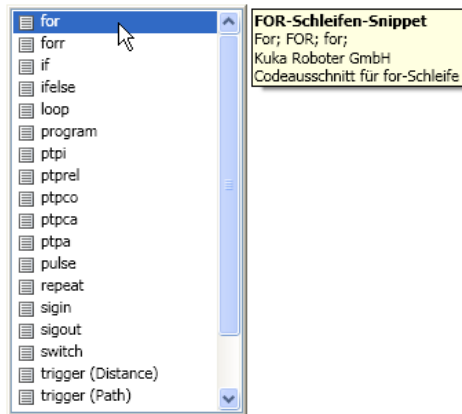


Рис. 18-46: Применение клавишей ввода или двойным щелчком

- Или Наберите сокращение и нажмите клавишу TAB.

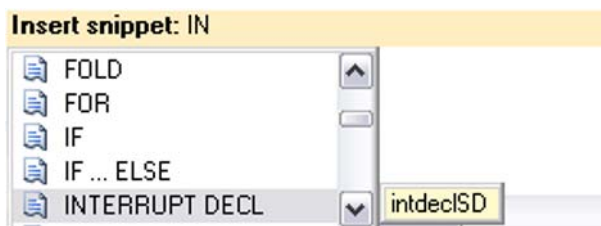


Рис. 18-47: Фрагменты кода KRL с функцией поиска

(Выяснить сокращения можно, вызвав поле списка с фрагментами кодов. При наведении указателя мыши на инструкцию сокращение отображается рядом с окном. Две последние буквы всегда SD, набирать их еще раз нельзя.)

2. Синтаксис KRL вставляется автоматически. Первое варьируемое место выделяется красным. Введите нужное значение.

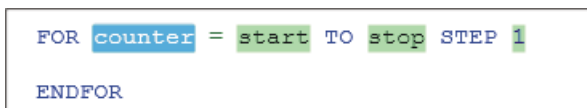


Рис. 18-48: Первое варьируемое место выделяется синим

3. Нажмите клавишу Enter для перехода к следующему варьируемому месту. Введите нужное значение.
4. Повторите шаг 3 для всех варьируемых мест.

## Формуляры

Содержимое редактора KRL можно, как и обычную программу KRL, структурировать с помощью формуляров.



Рис. 18-49: Закрытый формуляр

```
32 |
33 | ;fold outputs
34 |   $OUT[1]=true
35 |   $OUT[2]=true
36 |   $OUT[3]=true
37 | ;endfold (outputs)
38 |
39 |
```

**Рис. 18-50: Открытый формуляр**

Открытие формуляра:

Порядок действий

- Дважды нажмите на флажок закрытого формуляра.
- Или нажмите на знак «плюс».

Закрывание формуляра:

Порядок действий

- Нажмите на знак «минус».

Открытие или закрывание всех формуляров:

- Контекстное меню: **Формуляры > Открыть все формуляры** или **Заккрыть все формуляры**

## 19 Приложение

### 19.1 Сокращения

Для облегчения работы с системой управления KR C4 приведена выборка важнейших сокращений (акронимов) с краткими пояснениями.

Термин	Описание
CCU	<b>C</b> abinet <b>C</b> ontrol <b>U</b> nit
CCUsr	<b>C</b> abinet <b>C</b> ontrol <b>U</b> nit small robot
CIB	<b>C</b> abinet <b>I</b> nterface <b>B</b> oard
CIBsr	<b>C</b> abinet <b>I</b> nterface <b>B</b> oard small robot
CSP	<b>C</b> ontroler <b>S</b> ystem <b>P</b> anel (пульт системы контроллеров)
Карта Dual NIC	Двойная сетевая карта
EDS	<b>E</b> lectronic <b>D</b> ata <b>S</b> torage (карта памяти)
EMD	<b>E</b> lectronic <b>M</b> astering <b>D</b> evice (ранее EMT) для юстировки робота
EMV	<b>E</b> lektromagnetische <b>V</b> erträglichkeit (электромагнитная совместимость).
GBE	<b>С</b> еть <b>G</b> igabit <b>E</b> ther <b>N</b> et
KCB	<b>K</b> UKA <b>C</b> ontroller <b>B</b> us
KEB	<b>K</b> UKA <b>E</b> xension <b>B</b> us
KCP	<b>K</b> UKA <b>C</b> ontrol <b>P</b> anel (ручной программатор), новый термин: smartPAD
KLI	<b>K</b> UKA <b>L</b> ine <b>I</b> nterface
KOI	<b>K</b> UKA <b>L</b> ine <b>I</b> nterface
KPC	<b>K</b> UKA управляющий <b>П</b> К
KPP	<b>K</b> UKA <b>P</b> ower <b>P</b> ack
KPPsr	<b>K</b> UKA <b>P</b> ower <b>P</b> ack small robot
KRL	<b>K</b> UKA <b>R</b> obot <b>L</b> anguage (KUKA Roboter язык программирования)
KSB	<b>K</b> UKA <b>C</b> ontroller <b>B</b> us
KSP	<b>K</b> UKA <b>P</b> ower <b>P</b> ack
KSPsr	<b>K</b> UKA <b>P</b> ower <b>P</b> ack small robot
KSr	<b>K</b> UKA <b>C</b> ontroller <b>B</b> us
KSI	<b>K</b> UKA <b>L</b> ine <b>I</b> nterface
LWL	<b>L</b> icht- <b>W</b> ellen- <b>L</b> eiter (световод)
OPI	<b>O</b> perator <b>P</b> anel <b>I</b> nterface (подключение smartPAD)
PMB	<b>P</b> ower <b>M</b> anagement <b>B</b> oard
RCD	<b>R</b> esidual <b>C</b> urrent <b>D</b> evice; автомат защиты от тока утечки (FI)
RDC	<b>R</b> esolver <b>D</b> igital <b>C</b> onverter

Термин	Описание
SATA	<b>S</b> erial <b>A</b> dvanced <b>T</b> echnology <b>A</b> ttachment (шина обмена данными между процессором и жестким диском)
SIB	<b>S</b> afety <b>I</b> nterface <b>B</b> oard (интерфейсная плата безопасности)
SBC	<b>S</b> ingle <b>B</b> rake <b>C</b> ontroll
STO	<b>S</b> afe <b>T</b> orque <b>O</b> ff
SION	<b>S</b> afety <b>I</b> nput <b>O</b> utput <b>N</b> ode
USB	<b>U</b> niversal <b>S</b> erial <b>B</b> us (шинная система для соединения компьютера с периферийными устройствами)
USV	<b>Б</b> есперебойное <b>э</b> лектропитание

## 19.2 Используемые термины

STOP 0, STOP 1 и STOP 2 являются определениями останова согласно EN 60204-1:2006.

Термин	Описание
Зона оси	Зона действия каждой оси, указанная в градусах или миллиметрах, в пределах которой она может перемещаться. Зона перемещения оси должна определяться для каждой оси отдельно.
Остановочный путь	Остановочный путь = путь реакции + тормозной путь. Остановочный путь является частью опасной зоны.
Рабочая зона	В рабочей зоне манипулятор может передвигаться. Рабочая зона состоит из отдельных зон осей.
Пользователь (оператор)	Пользователем промышленного робота может быть предприниматель, работодатель или уполномоченное лицо, несущее ответственность за эксплуатацию промышленного робота.
Опасная зона	Опасная зона включает в себя рабочую зону и остановочные пути.
Срок службы	Срок службы важных для безопасности деталей начинается с момента ее поставки заказчику. На продолжительность срока службы не влияет ни место эксплуатации детали (система управления роботом и другое место), ни время ввода в эксплуатацию, так как важные для безопасности детали подвергаются старению и во время хранения.
KCP	Ручной программатор KCP (KUKA Control Panel) оснащен всеми функциями управления и индикации, необходимыми для управления промышленным роботом и его программирования. Вариант пульта KCP для системы управления KR C4 называется KUKA smartPAD. Как правило, в этой документации применяется общее обозначение KCP.
KRF	<b>K</b> ontrollierte <b>R</b> oboterfahrt (Контролируемый Ход Робота) Режим работы KRF доступен только в том случае, если используются программы KUKA.SafeOperation и KUKA.SafeRangeMonitoring. Если робот нарушил элемент контроля и останавливается системой управления безопасностью, то его можно вывести из нарушенной зоны в режиме KRF.
Манипулятор	Механика робота и соответствующее электрооборудование
Безопасная зона	Безопасная зона находится за пределами опасной зоны.

Термин	Описание
Безопасный рабочий останов	<p>Безопасный рабочий останов – это функция контроля останова. Он не останавливает перемещение самого робота, а контролирует состояние останова осей робота. Если оси перемещаются при безопасном рабочем останове, срабатывает безопасный останов STOP 0.</p> <p>Безопасный рабочий останов также можно выполнить внешними средствами.</p> <p>При срабатывании безопасного рабочего останова система управления роботом задействует выход к магистральной шине. Выход также задействуется в том случае, если к моменту срабатывания остановлены не все оси, что ведет к срабатыванию безопасного останова STOP 0.</p>
Безопасный останов STOP 0	<p>Останов, который выполняется по команде системы управления безопасностью. Система управления безопасностью мгновенно отключает приводы и подачу питания к тормозам.</p> <p><b>Указание:</b> этот останов обозначается в настоящей документации в качестве безопасного останова 0.</p>
Безопасный останов STOP 1	<p>Останов, который выполняет и контролирует система управления безопасностью. Процесс торможения выполняется компонентом системы управления роботом, который не относится к безопасности, и контролируется системой управления безопасностью. Как только манипулятор останавливается, система управления безопасностью отключает приводы и подачу питания к тормозам.</p> <p>При срабатывании безопасного останова STOP 1 система управления роботом задействует выход к магистральной шине.</p> <p>Безопасный останов STOP 1 также можно выполнить внешними средствами.</p> <p><b>Указание:</b> этот останов обозначается в настоящей документации в качестве безопасного останова 1.</p>
Безопасный останов STOP 2	<p>Останов, который выполняет и контролирует система управления безопасностью. Процесс торможения выполняется компонентом системы управления роботом, который не относится к безопасности, и контролируется системой управления безопасностью. Приводы остаются включенными, а тормоза – разомкнутыми. Сразу после остановки манипулятора срабатывает безопасный рабочий останов.</p> <p>При срабатывании безопасного останова STOP 2 система управления роботом задействует выход к магистральной шине.</p> <p>Безопасный останов STOP 2 также можно выполнить внешними средствами.</p> <p><b>Указание:</b> этот останов обозначается в настоящей документации в качестве безопасного останова 2.</p>
Опции безопасности	<p>Общее название для опций, которые обеспечивают возможность конфигурации функций контроля в дополнение к стандартным функциям безопасности.</p> <p>Пример: SafeOperation</p>
Категория останова 0	<p>Приводы незамедлительно отключаются, после чего срабатывают тормоза. Манипулятор и дополнительные оси (опция) останавливаются вблизи траектории перемещения.</p> <p><b>Указание:</b> данная категория останова обозначается в документации как STOP 0.</p>

Термин	Описание
Категория останова 1	<p>Манипулятор и дополнительные оси (опция) останавливаются точно на траектории перемещения.</p> <ul style="list-style-type: none"> <li>■ Режим работы T1: происходит отключение приводов сразу после останова робота, но не позднее чем через 680 мс.</li> <li>■ Режимы работы T2, AUT, AUT EXT: происходит отключение приводов через 1,5 секунды.</li> </ul> <p><b>Указание:</b> данная категория останова обозначается в документации как STOP 1.</p>
Категория останова 2	<p>Приводы не отключаются, а тормоза не срабатывают. Манипулятор и дополнительные оси (опция) останавливаются со временем торможения согласно траектории перемещения.</p> <p><b>Указание:</b> данная категория останова обозначается в документации как STOP 2.</p>
Системный интегратор (интегратор установок)	<p>Системные интеграторы – это лица, которые с учетом требований техники безопасности интегрируют промышленный робот в установку, а также осуществляют его ввод в эксплуатацию.</p>
T1	<p>Тестовый режим работы «Вручную, пониженная скорость» (&lt;= 250 мм/с)</p>
T2	<p>Тестовый режим работы «Вручную, высокая скорость» (допускается &gt; 250 мм/с)</p>
Дополнительная ось	<p>Ось движения, не являющаяся частью манипулятора, однако управляемая системой управления роботом, например, линейный блок KUKA, поворотный откидной стол, Posiflex.</p>

### 19.3 Выдержки из описания «Безопасность KR C4»

В данном разделе только фрагментарно рассматриваются некоторые темы из описания «Безопасность KUKA KR C4». Они в основном касаются планирования.

#### Обзор предохранительных устройств

- Выбор режимов работы
- Защита оператора (= соединение для блокировки заграждающих защитных приспособлений)
- Устройство АВАРИЙНОГО ОСТАНОВА
- Устройство подтверждения
- Внешний безопасный рабочий останов
- Внешний безопасный останов 1 (кроме системы управления варианта «KR C4 compact»)
- Внешний безопасный останов 2
- Контроль скорости в T1

Функции безопасности промышленного робота соответствуют следующим требованиям:

- категория 3 и Performance Level d согласно EN ISO 13849-1:2008;

Однако данные требования выполняются только при соблюдении следующего условия:

- устройство АВАРИЙНОГО ОСТАНОВА приводится в действие не реже одного раза в 6 месяцев.

**ОПАСНО** Без функционирующих предохранительных устройств и защитных приспособлений промышленный робот может причинить травмы и материальный ущерб. Если предохранительные устройства и защитные приспособления деактивированы или демонтированы, эксплуатация промышленного робота запрещена.

**!** При планировании установки следует дополнительно учитывать и рассчитывать функции безопасности всей установки. Следует интегрировать промышленный робот в систему безопасности всей установки.

### Система управления безопасностью

Система управления безопасностью – устройство внутри управляющего ПК. Она сочетает в себе сигналы и системы контроля, относящиеся к безопасности.

Задачи системы управления безопасностью:

- выключение приводов, приведение в действие тормозов;
- контроль времени торможения;
- контроль простоя (после останова);
- контроль скорости в T1
- оценка сигналов, относящихся к безопасности;
- установка выходов, относящихся к безопасности.

### Выбор режимов работы

Промышленный робот может эксплуатироваться в следующих режимах работы:

- режим «Вручную, пониженная скорость» (T1);
- режим «Вручную, высокая скорость» (T2);
- автоматический режим (AUT);
- режим внешней автоматики (AUT EXT).
- KRF

**i** Не изменять режим работы во время обработки программы. При изменении режима работы во время обработки программы промышленный робот остановится в режиме безопасного останова 2.

Режим работы	Использование	Скорости
T1	для режима тестирования, программирования и обучения;	<ul style="list-style-type: none"> <li>■ Верификация программы: запрограммированная скорость, максимум 250 мм/с</li> <li>■ ручной режим: скорость перемещения вручную, максимум 250 мм/с</li> </ul>
T2	Для режима тестирования	<ul style="list-style-type: none"> <li>■ Верификация программы: запрограммированная скорость</li> <li>■ ручной режим: невозможен</li> </ul>
AUT	для промышленных роботов без системы управления верхнего уровня;	<ul style="list-style-type: none"> <li>■ Программный режим: запрограммированная скорость</li> <li>■ ручной режим: невозможен</li> </ul>



Режим работы	Использование	Скорости
AUT EXT	Для промышленных роботов с системой управления верхнего уровня, например, ПЛК	<ul style="list-style-type: none"> <li>■ Программный режим: запрограммированная скорость</li> <li>■ ручной режим: невозможен</li> </ul>
KRF	<p>KRF доступно только в том случае, если используется KUKA.SafeOperation или KUKA.SafeRangeMonitoring.</p> <p>Если робот нарушил элемент контроля и останавливается системой управления безопасностью, то его можно вывести из нарушенной зоны в режиме KRF.</p> <p>Скорости соответствуют режиму T1</p>	

**Система защиты оператора**

- Сигнал «Защита оператора» служит для блокировки разделительных защитных приспособлений, таких как, например, защитные двери.
- Без этого сигнала невозможен автоматический режим работы.
- При потере сигнала во время работы в автоматическом режиме (например, при открытии защитных дверей) манипулятор переключается в режим безопасного останова 1.
- В режимах «Вручную, пониженная скорость» (T1), «Вручную, высокая скорость (T2)» и KRF система защиты оператора не активизирована.

**⚠ ОСТОРОЖНО** После потери сигнала запрещается продолжать работу в автоматическом режиме только посредством закрывания защитного приспособления; помимо этого требуется квитирование. Об этом должен позаботиться системный интегратор. Это предотвращает возобновление работы в автоматическом режиме по неосторожности, когда в опасной зоне находятся люди, например, при закрывании защитной двери.

- Квитирование должно быть организовано таким образом, чтобы предварительно проводилась фактическая проверка опасной зоны. Квитирование, которое не позволяет этого (например, из-за того, что оно автоматически следует за закрыванием защитного устройства), является недопустимым.
- Несоблюдение данного предупреждения может привести к гибели людей, тяжелым телесным повреждениям или значительному материальному ущербу.

**Устройство АВАРИЙНОГО ОСТАНОВА**

- Устройством АВАРИЙНОГО ОСТАНОВА промышленного робота является устройство АВАРИЙНОГО ОСТАНОВА на КСР.
- Данное устройство необходимо задействовать в опасной или аварийной ситуации.
- Реакция промышленного робота при задействовании устройства аварийного останова: манипулятор и дополнительные оси (опция) переключаются в режим безопасного останова 1.
- Для продолжения работы устройство аварийного останова следует деблокировать посредством вращения.
- Следует установить не менее одного внешнего устройства АВАРИЙНОГО ОСТАНОВА. В этом случае устройство аварийного останова доступно даже при отключенном устройстве КСР.

**⚠ ОСТОРОЖНО** Если от инструментов или других устройств, соединенных с манипулятором, может исходить опасность, необходимо интегрировать их в контур АВАРИЙНОГО ОСТАНОВА на установке.  
Несоблюдение данного предупреждения может привести к гибели людей, тяжелым травмам или значительному материальному ущербу.

### Внешнее устройство аварийного останова

- На каждой станции управления, которая может выполнить перемещение робота или создать другую опасную ситуацию, должны быть установлены устройства аварийного останова.
- За это несет ответственность системный интегратор.
- Следует установить не менее одного внешнего устройства АВАРИЙНОГО ОСТАНОВА. В этом случае устройство аварийного останова доступно даже при отключенном устройстве КСР.
- Внешние устройства АВАРИЙНОГО ОСТАНОВА подключаются через интерфейс пользователя.
- Внешние устройства АВАРИЙНОГО ОСТАНОВА не входят в комплект поставки промышленного робота.

### Устройство подтверждения

Устройство подтверждения промышленного робота представляет собой переключатели подтверждения на КСР.

На КСР расположены три переключателя подтверждения. Каждый из переключателей подтверждения имеет три положения:

- не нажат;
- среднее положение;
- нажат до отказа (экстренное положение).

В тестовых режимах работы и режиме KRF перемещение манипулятора возможно лишь при удержании переключателя подтверждения в среднем положении.

- При отпускании переключателя подтверждения сработает безопасный останов 2.
- При нажатии переключателя подтверждения до отказа сработает безопасный останов 1.
- В течение короткого времени можно одновременно удерживать два переключателя подтверждения в среднем положении. Это обеспечивает переход от одного переключателя подтверждения к другому. Если два переключателя подтверждения одновременно удерживаются в течение долгого времени в среднем положении, через несколько секунд сработает безопасный останов.

**⚠ ОСТОРОЖНО** Запрещается фиксировать переключатель подтверждения клейкой лентой или прочими вспомогательными средствами, а также воздействовать на него иным способом.  
Существует опасность получения травм, в том числе и смертельных, а также опасность нанесения материального ущерба.

### Обзор режимов работы и защитных функций

В приведенной ниже таблице указано, в каком режиме работы применяются защитные функции.

Защитные функции	T1, KRF	T2	AUT	AUT EXT
Система защиты оператора	-	-	активир.	активир.
Устройство АВАРИЙНОГО ОСТАНОВА	активир.	активир.	активир.	активир.
Устройство подтверждения	активир.	активир.	-	-
Пониженная скорость при верификации программы	активир.	-	-	-
Пошаговый режим	активир.	активир.	-	-
Программируемые концевые выключатели	активир.	активир.	активир.	активир.

**Возможности перемещения манипулятора без системы управления роботом**

Для того чтобы после аварийной ситуации или поломки переместить манипулятор вручную, можно воспользоваться следующими приспособлениями/возможностями:

- Устройство свободного вращения (опция)  
Устройство свободного вращения также может быть использовано для приводных двигателей основной оси и для приводных двигателей оси руки в зависимости от исполнения робота.



**Рис. 19-1: Устройство свободного вращения**

- Устройство размыкания тормозов (опция)  
Устройство размыкания тормозов предусмотрено для моделей роботов, к двигателям которых отсутствует свободный доступ.  
Пример для робота AGILUS:



**Рис. 19-2: Пульт ручного управления**

- 1 Пульт ручного управления      2 Клавиша управления S1

- Перемещение осей руки вручную  
Оси руки моделей роботов с низкой грузоподъемностью не оснащены устройством свободного вращения. В использовании данного устройства нет необходимости, так как оси руки можно перемещать вручную.

Данные приспособления/возможности разрешается применять только в исключительных случаях и в аварийных ситуациях, например, для освобождения людей.



Информацию о том, какие возможности предлагаются для конкретных моделей роботов и как они используются, можно найти в инструкции по монтажу или в инструкции по эксплуатации соответствующего робота или запросить в компании KUKA Roboter GmbH.



Во время эксплуатации двигателя достигают высоких температур, в связи с чем возникает опасность получения ожогов кожи. Не касаться двигателей! Принять соответствующие меры безопасности, например, надеть защитные перчатки.



## Индекс

### А

Абсолютная точность 47  
 Аварийный останов 68  
 Аварийный останов, внешний 285  
 Администраторы 175  
 Активация проекта 263

### Б

Безопасная зона 280  
 Безопасность KR C4 282  
 Безопасный останов 0 281  
 Безопасный останов 1 281  
 Безопасный останов 2 281  
 Безопасный останов STOP 0 281  
 Безопасный останов STOP 1 281  
 Безопасный останов STOP 2 281  
 Безопасный рабочий останов 281  
 Бесконечный цикл 180  
 Блок управления 52  
 Блокировка разделительных защитных приспособлений 284

### В

Вес 51  
 Виброустойчивость 52  
 Внешние датчики безопасности 38  
 Внешняя автоматика 227, 231, 232  
 Внешняя система подачи энергии 31  
 Время жизни 191  
 Встроенная в робот система подачи энергии 32  
 Встроенный формуляр 138  
 Выбор программы 126  
 Выбор режимов работы 282, 283  
 Выравнивание потенциалов 52  
 Высота установки 52

### Г

Габаритные размеры системы управления роботом 53  
 Гибкость 7  
 Главные оси 46  
 глобально 191  
 Глубокая разрядка аккумулятора 52  
 Грузоподъемность 30  
 Группа пользователей, по умолчанию 175

### Д

Данные дополнительной загрузки (пункт меню) 108  
 Данные загрузки инструмента 102  
 Данные загрузки инструмента (пункт меню) 106  
 Датчик 33  
 Датчики 33  
 Декартовы координаты 45  
 Диалоговое сообщение 71  
 Директива VDI 2860 41

Длина кабелей 53  
 Дополнительная нагрузка 28, 30  
 Дополнительные оси 282  
 Допуск для номинального напряжения питающей сети 51

### З

Заводская табличка 70  
 Загрузка проекта 255  
 Законы робототехники 8  
 Закрепление 265  
 Запрос IF 183  
 Запуск программы 126  
 Запуск программы от ПЛК 227  
 Защита оператора 282  
 Защитные ограждения 37  
 Защитные функции 286  
 Зона оси 280  
 Зонный контроль за счет разделения на рабочие зоны 35

### И

Изменение, команды перемещения 155  
 Импорт, проект WorkVisual 266  
 Инициализация 125, 197  
 Инструмент 31  
 Интегратор установок 282  
 Информационное сообщение 71  
 Исполнительный орган 31

### К

Кабельные удлинители пульта управления smartPAD 53  
 Калибровка базы 116  
 Калибровка инструмента 108  
 Карта Dual NIC 279  
 Каталоги (окно) 253  
 Категория останова 0 281  
 Категория останова 1 282  
 Категория останова 2 282  
 Квитируемое сообщение 71  
 Клавиатура 69  
 Клавиша клавиатуры 69  
 Клавиша обратного пуска 69  
 Клавиша пуска 69, 70  
 Клавиша СТОП 69  
 Клавиши перемещения 68  
 Клавиши состояния 69  
 Класс влажности 52  
 Класс защиты 51  
 Ключевое слово 194  
 Команда, обусловленная 183  
 Конструкции шарниров 46  
 Контроль выполнения программы 179  
 Контроль ориентации 149, 155  
 Координаты вертикальных шарниров 45  
 Координаты горизонтальных шарниров 45  
 Координаты формы цилиндра 45

Координаты формы шара 45

## Л

Линейные оси 47  
 Логика, общая информация 161  
 Логические операции 199  
 локально 191

## М

Манипулирование заготовками 13  
 Манипулятор 280  
 Манипуляция 199  
 Массивы 202  
 Менеджер соединений 68  
 Механические концевые упоры 34  
 Минимальные расстояния для системы управления роботом 54  
 Монтаж 14

## Н

Нагрузки 28  
 Нагрузки на роботе 101  
 Неотклоняемый цикл 182  
 Номинальная нагрузка 28  
 Номинальное напряжение питающей сети 51

## О

Обзор приложений и шинных систем 59  
 Обзор системы управления роботом 55  
 Обзор шинных систем 59  
 Область применения 30  
 Объявление 194, 195  
 Опасная зона 280  
 Оператор 280  
 Операторы 175  
 Операции сравнения 199  
 Описания 191  
 Опции безопасности 281  
 Оси вращения 47  
 Основание робота 79  
 Основная система координат 79  
 Основные арифметические действия 199  
 Основные характеристики 51  
 Остановочный путь 280  
 Отклоняемый цикл 181  
 Открытие проекта 251

## П

Панель управления WorkVisual 252  
 Передача параметров 189  
 Передача проекта в систему управления роботом (установка) 259  
 Переключатели подтверждения 285  
 Переключатель подтверждения 70  
 Переменные 191, 194  
 Перемещение вручную по осям 74  
 Перемещение вручную, база 89  
 Перемещение вручную, инструмент 86  
 Перемещение вручную, универсальное 81  
 Перемещение осей по отдельности 74  
 Перемещение CIRC 145

Перемещение LIN 145  
 Перемещение PTP 139  
 Переход 179  
 Переход многократный 183  
 Переход условный 183  
 Перечислимый тип данных 208  
 Побитовые операции 199  
 Подача энергии 31  
 Подключение к сети, технические характеристики 51  
 Подпрограмма 185  
 Подпрограмма, глобальная 187  
 Подпрограмма, локальная 185  
 Пользователь 280  
 Поля 202  
 Портальные роботы 45  
 Пошаговый режим 286  
 Предохранитель со стороны питающей сети 51  
 Предохранительные устройства 33  
 Предохранительные устройства, обзор 282  
 Применение промышленных роботов 13  
 Принудительно состыкованные оси 47  
 Приоритет 201  
 Проводник проектов 254  
 Программа дублирование 135  
 Программа переименование 135  
 Программа удаление 135  
 Программирование перемещений робота 137  
 Программирование перемещений с помощью KRL 211  
 Программируемые концевые выключатели 34, 286  
 Программисты 175  
 Продолжительность такта 30  
 Продуктивность 7  
 Промышленные роботы 8, 9  
 Пространственная мышь 68  
 Пульт управления KUKA 67  
 Пульт управления KUKA smartPAD 53  
 Путь реакции 280

## Р

Рабочая зона 30, 280  
 Рабочие зоны (окно) 253  
 Различение ситуаций 183  
 Разная длина кабелей синусно-косинусного преобразователя 53  
 Разъем USB 70  
 Распределитель 183  
 Редактор KRL 272  
 Режим интерполяции 144, 154  
 Режим работы 72  
 Робототехническая система 27  
 Робототехническая ячейка 27  
 Роботы 7, 8, 40  
 Роботы Scara 45  
 Россумские универсальные роботы 8

## С

Свойства (окно) 253

- Сглаживание 144, 155  
 Сглаживание CIRC 151  
 Сглаживание LIN 151  
 Сглаживание PTP 141  
 Сенсорный экран 67  
 Сетевая частота 51  
 Сингулярность 146  
 Система защиты оператора 284, 286  
 Система координат 79  
 Система координат инструмента 79  
 Система координат фланца 79  
 Система управления безопасностью 283  
 Система управления, обзор 55  
 Системная шина KUKA, KSB 62  
 Системный интегратор 282  
 Смена инструмента 13  
 Сокращения 279  
 Сообщение о состоянии 71  
 Сообщение об ожидании 71  
 Сообщения 70  
 Сообщения (окно) 253  
 Справка 252  
 Сравнение проектов 255  
 Срок службы 280  
 Стабильность повторяемости 47  
 Стандартные функции 199  
 Строка меню 252  
 Строки кнопок 252  
 Структура 205  
 Структура проекта (окно) 253
- Т**  
 Температура окружающей среды 52  
 Термины, безопасность 280  
 Тип шкафа 51  
 Ток полной нагрузки 51  
 Тормозной путь 280  
 Точка запястья 147
- У**  
 Угол круга 215  
 Указатель предварительного выполнения 194  
 Универсальная система координат 79  
 Уровень шума 51  
 Уровень эксперта 175  
 Условия окружающей среды 52  
 Условия эксплуатации 28  
 Устройство АВАРИЙНОГО ОСТАНОВА 284, 286  
 Устройство аварийного останова 285  
 Устройство подтверждения 285, 286  
 Устройство размыкания тормозов 286  
 Устройство свободного вращения 77, 286
- Ф**  
 Функции 185  
 Функции переключения, простые 167  
 Функции, математические 190  
 Функция 189  
 Функция ожидания 163
- Функция переключения, траектория 170
- Х**  
 Хранение данных 191
- Ц**  
 Цикл с подсчетом 180  
 Цикл FOR 180  
 Цикл REPEAT 182  
 Цикл WHILE 181  
 Цикл бесконечный 180  
 Цикл неотклоняемый 182  
 Цикл отклоняемый 181  
 Цикл с подсчетом 180  
 Циклы 179, 180
- Ч**  
 Чапек Карел 8
- Ш**  
 Шаблон 266  
 Шина контроллера KUKA, KCB 61
- Э**  
 Экстренное положение 76, 285  
 Эффективность использования энергии 65
- Я**  
 CCU 279  
 CCUsr 279  
 CIB 279  
 CIBsr 279  
 CSP 279  
 DECL 194  
 EDS 279  
 EMD 279  
 EMV 279  
 EN 10218-1 41  
 EN 775 41  
 ENUM 208  
 GBE 279  
 IF -ELSE 183  
 KCB 279  
 KCP 67, 279, 280  
 KEB 279  
 KLI 279  
 KOI 279  
 KPC 279  
 KPP 279  
 KRF 280  
 KRL 279  
 KSB 279  
 KSI 279  
 KSP 279  
 KUKA Extension Bus, KEB 63  
 KUKA Line Interface, KLI 64  
 KUKA smartPAD 67, 280  
 LOOP 180  
 LWL 279  
 OPI 279  
 Performance Level 282



PMB 279  
R.U.R. 8  
RCD 279  
RDW 279  
Return 186  
SAK 125  
SATA 280  
SBC 280  
SIB 280  
SION 280  
smartPAD 67, 280  
STO 280  
STOP 0 280, 281  
STOP 1 280, 282  
STOP 2 280, 282  
SWITCH - CASE 183  
T1 282  
T2 282  
Unimate 10  
USB 280  
USV 280  
WAIT 163  
WAIT FOR 164  
WorkVisual 243, 251

