

ПРИМЕР КОНКУРСНОГО ЗАДАНИЯ



ПО КОМПЕТЕНЦИИ «ПРОМЫШЛЕННАЯ РОБОТОТЕХНИКА»

ОГЛАВЛЕНИЕ

1. ОБЩЕЕ ВРЕМЯ НА ВЫПОЛНЕНИЕ ЗАДАНИЯ: 18 Ч.....	3
2. КОНКУРСНОЕ ЗАДАНИЕ	3
3. Построение модели стенда в ПО «Рациональное производство»	3
4. Описание виртуальной ячейки и начальных условий.	4
5. Алгоритм работы.....	4
6. Дополнительные требования к программе	5
7. Техника безопасности.....	6
8. Распознавание столкновений	6
9. Post-Processor для выгрузки программ на работа	7
10. Подготовка робота и создание двойника	8
11. Режимы работы робота KUKA	10
12. Данные для заполнения TOOL и BASE	11
13. Карта сигналов робота.....	11
14. Общие правила именования в KRL (KUKA Robot Language)	12
15. Результаты	12



1. **ОБЩЕЕ ВРЕМЯ НА ВЫПОЛНЕНИЕ ЗАДАНИЯ: 18**

Ч.

2. **КОНКУРСНОЕ ЗАДАНИЕ**

Задание включает в себя офлайн программирование производственного оборудования, виртуальную пуско-наладку и запуск разработанных в цифровом двойнике программ на реальном оборудовании посредством передачи команд оператору.

Программирование производственного оборудования и виртуальная пуско-наладка в ПО «Рациональное производство».

Команде необходимо:

- Разработать цифровой двойник роботизированной производственной ячейки (или линии) в среде «Рациональное производство»;
- Запрограммировать реализуемые ячейкой (линией) производственно-технологические процессы в среде «Рациональное производство»;
- Произвести виртуальную пуско-наладку ячейки (линии);
- Произвести пуско-наладку ячейки (линии) на реальном стенде посредством связи с оператором.

3. **ПОСТРОЕНИЕ МОДЕЛИ СТЕНДА В ПО «РАЦИОНАЛЬНОЕ ПРОИЗВОДСТВО»**

Параметры захвата, продуктов и подставок для продуктов настроены в предоставленных компонентах, форму этих моделей можно не изменять. В дополнение к компонентам участникам выдается файл с ячейкой, где произведена настройка положения робота на столе, все остальные компоненты необходимо переместить, либо настроить свойства, если этого требует корректное выполнение программы.

На реальном роботе уже проведена калибровка инструмента (tool) и базовой системы координат (base), настроены входные и выходные сигналы. Необходимо учесть эти данные в виртуальной ячейке.



4. ОПИСАНИЕ ВИРТУАЛЬНОЙ ЯЧЕЙКИ И НАЧАЛЬНЫХ УСЛОВИЙ.

На рабочем столе расположены две зоны. В каждой зоне находится подставка для продуктов (цилиндров). Перед каждой подставкой установлен датчик присутствия, подключенный к системе управления робота:

- Датчик зоны 1 подключен к входному сигналу IN_267 робота.
- Датчик зоны 2 подключен к входному сигналу IN_265 робота.

Исходное состояние:

- В ячейке находится 1 цилиндр. Он может располагаться в подставке зоны 1 или в подставке зоны 2.
- Оси робота находятся в начальном положении: $A1 = 0^\circ$, $A2 = -90^\circ$, $A3 = 90^\circ$, $A4 = 0^\circ$, $A5 = 0^\circ$, $A6 = 0^\circ$. Из этого положения ячейка должна начать свою работу.

5. АЛГОРИТМ РАБОТЫ.

Участникам необходимо разработать программу с бесконечным циклом, которая выполняет следующую последовательность действий:

- Приведение робота в начальное положение. Программа должна гарантированно возвращать робота в указанную стартовую позицию в начале каждого цикла.
- Проверка наличия цилиндра в подставке зоны 1.
- Программа считывает состояние датчика IN_267.
- Если цилиндр присутствует в зоне 1, робот должен переложить его из подставки зоны 1 в подставку зоны 2.
- Проверка наличия цилиндра в подставке зоны 2 (при условии пустой зоны 1).



- Если в подставке зоны 1 цилиндра нет, программа считывает состояние датчика IN_265.

- Если цилиндр присутствует в зоне 2, робот должен переложить его из подставки зоны 2 в подставку зоны 1.

- После выполнения операции переукладывания или если цилиндр отсутствует в обеих зонах, программа возвращается к шагу 1, начиная новый цикл.

6. ДОПОЛНИТЕЛЬНЫЕ ТРЕБОВАНИЯ К ПРОГРАММЕ

- Структура программы. При написании программы необходимо разбить программу на подпрограммы.

- Сопроводительные комментарии. Для облегчения дальнейшей работы с программой необходимо предусмотреть пояснительные и документирующие комментарии.

- Счетчик перенесенных продуктов. Необходимо зарегистрировать переменную, в которой будет храниться информация о количестве деталей (каждой отдельной), которые перенес робот в течении выполнения программы.

- Проверка захвата. Ваша программа должна предусматривать, что в начале ее выполнения захват может быть закрыт, следовательно в начале выполнения каждой программы необходимо подать сигнал на открытие захвата.

4. Произвести перевод программы на язык KRL посредством использования инструмента «Постпроцессор». Итогом должна быть программа, успешно выполняющая свою функцию на реальном роботе.

5. Произвести тестирование и отладку программы. Итогом должна быть программа, успешно выполняющая свою функцию на реальном роботе без столкновений с внешним оборудованием и рабочими поверхностями.



5.1. Для тестирования нужно передать программу одному из экспертов на площадке (два файла с одинаковым названием и расширениями .src, .dat).

4.2 К программе необходимо приложить текстовый файл с комментариями для оператора. В этом файле указать режим работы, вид работы, коррекцию скорости выполнения программы. Дополнительно можно запросить у оператора данные с робота (актуальную позицию в нужном положении робота, текущее состояние сигнала), либо сообщить полезную информацию (позиции, где возможны столкновения, необходимость контролировать кабели с воздухом и пр.).

6. Передать конечные файлы для проверки экспертами. Конечные файлы состоят из: двух файлов планировки в формате «.grpo» (одна планировка с виртуальной ячейкой, вторая – та, что использовалась для программирования и отладки), два файла программы (с расширениями «.src», «.dat»)

7. ТЕХНИКА БЕЗОПАСНОСТИ

При работе с ячейкой необходимо учесть следующий порядок отладки программы:

1. Запуск робота в тестовом режиме T1, пошагово, с коррекцией скорости.
2. Запуск робота в тестовом режиме T1
3. Запуск робота в режиме AUT.

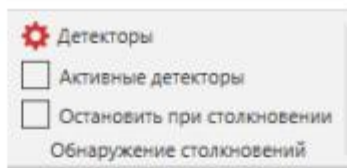
В конечных файлах программы, которые вы передаете для проверки, параметры скорости должны быть следующими: осевой тип перемещения – 15%, движение по траектории – 0.1 м/с.

8. РАСПОЗНАВАНИЕ СТОЛКНОВЕНИЙ

Распознавание столкновений позволяет устранить возможные столкновения еще на этапе программирования внутри P-Про. Это позволяет избежать чрезвычайных ситуаций, а также экономит время на тестирование программ



на реальном оборудовании, что снижает издержки на простой ячейки. R-Pro имеет следующий функционал:



Команда	Описание
Детекторы	<p>Отображает список опций и инструментов для управления проверками на столкновение.</p> <p><i>Обнаружить столкновение</i> Обнаружение первого или всех столкновений.</p> <p><i>Допуск столкновения</i> Обнаружение столкновения при ударе или в пределах допустимого расстояния.</p> <p><i>Показать минимальное расстояние</i> При использовании устойчивости к столкновению показать/скрыть кратчайшее расстояние между объектами при обнаруженном столкновении.</p>
	<p><i>Выделенное против Мирового</i> Определите, сталкивается ли выбранный компонент с любым другим компонентом в 3D-мире.</p> <p><i>Создать детектор</i> Создает новый детектор столкновений. Дополнительные параметры отображаются на панели задач, доступ к которой можно получить, щелкнув детектор в списке. Щелкните компонент в трехмерном мире, а затем используйте мини-панель инструментов или панель задач, чтобы добавить компонент в список А или В в детекторе. На панели задач используйте вкладки А и В и флажки узлов, чтобы включить/исключить узлы из детектора.</p>
Активные детекторы	Включение/выключение всех детекторов при проверке на столкновение.
Остановить при столкновении	Остановка запущенной симуляции при обнаружении столкновения.

9. POST-PROCESSOR ДЛЯ ВЫГРУЗКИ ПРОГРАММ НА РОБОТА

Вам будет представлен .zip файл – Post-Processor для выгрузки программ на робота. Его необходимо распаковать по следующему пути:
 C:\Users*Ваше имя пользователя*\Documents\R-Pro\0.2\My Commands

Ознакомьтесь с описанием команд Пост-Процессора



Команда	Описание
Post Processor	Автоматически подстраивается под выбранного робота. Определяет формат выходного файла
Output	Место, куда сохраняется программа
Use spline motions	Использование новых типов движения (SLIN/SPTP)
Use inline form	Использование структуры Folds
Comment out Base/Tool	Комментирование баз/инструментов вместо их определения внутри программы

* Так как базы и инструменты уже настроены на ячейке, вам не нужно определять их в рабочей программе на работе (иначе говоря, нужно использовать команду «Comment out Base/Tool»)

10. ПОДГОТОВКА РОБОТА И СОЗДАНИЕ ДВОЙНИКА

Перед программированием робота в Р-Про необходимо сделать следующее:

1. Добавление и калибровка инструмента на работе
2. Добавление и калибровка базы на работе

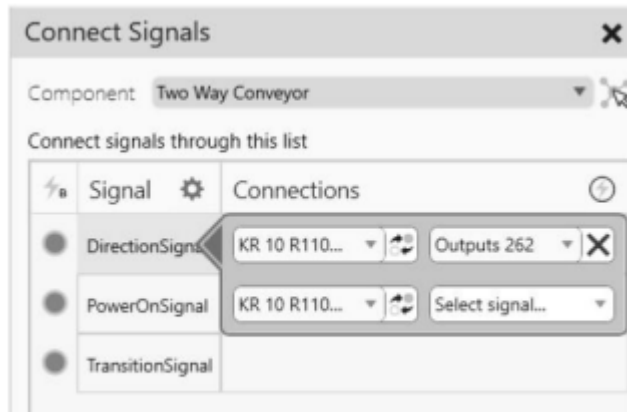
* Шаги 1, 2 уже выполнены, данные калибровки предоставляются участникам ниже

3. Добавление соответствующих инструментов и баз на работе в Р-Про на основе предоставленных начальных данных

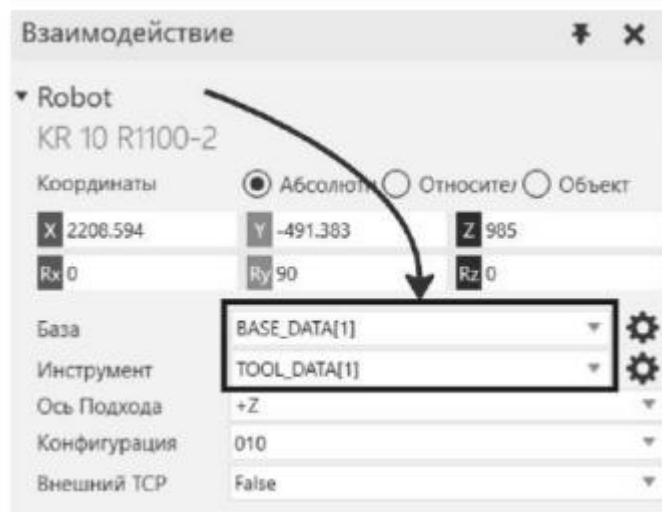
4. Подключение сигналов

Важно понимать, что реальное оборудование может иметь погрешность (в том числе формы и расположения), поэтому необходимо тестирование написанной оффлайн программы на пониженных скоростях даже после подготовки двойника ячейки.

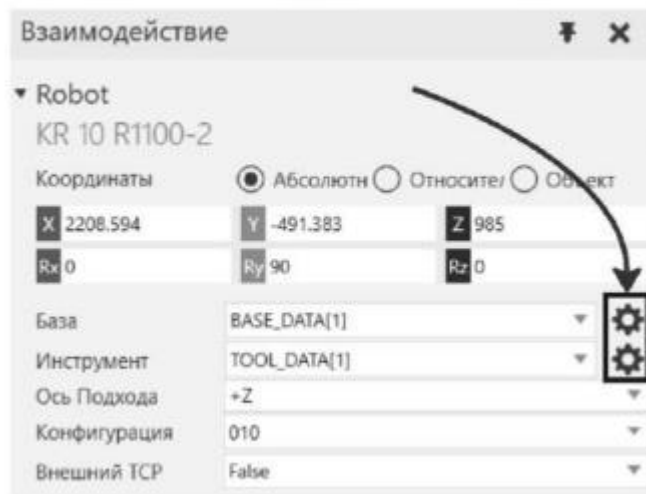




Подключение сигналов



Калибровка инструмента и базы.



BASE_DATA[1]		
Координаты <input checked="" type="radio"/> Абсолютн <input type="radio"/> Относител <input type="radio"/> Объект		
X	1578.594	Y -491.383
Z	0	
Rx	0	Ry 0
Rz	0	
Наименование	BASE_DATA[1]	
Преобразовать		
Узел	*WORLD*	
РежимИнтерполя...	#NONE	
Description		
AttachedKinematic		

Добавление соответствующих инструментов и баз на работе в Р-Про на основе предоставленных начальных данных

11. РЕЖИМЫ РАБОТЫ РОБОТА КУКА

• T1 (Ручной режим, пониженная скорость)

- Для тестирования, программирования, обучения
- Максимальная скорость во всех режимах 250 мм/с

• T2

- Для тестирования, программирования, обучения
- Скорость соответствует запрограммированной
- Перемещение робота вручную невозможно

• AUT

- Для промышленных роботов без системы управления верхнего уровня
- Скорость соответствует запрограммированной
- Перемещение робота вручную невозможно

• EXT AUT

- Для промышленных роботов с системой управления верхнего уровня
- Скорость соответствует запрограммированной



- Перемещение работа вручную невозможно

12. ДАННЫЕ ДЛЯ ЗАПОЛНЕНИЯ TOOL и BASE

TOOL[7]={X 6.18, Y 39.74, Z 123.54, A 0.0, B 0.0, C 0.0} – система начала

координат расположена в углу пальца захватного устройства

BASE[6]={X 577.74, Y -448.14, Z -12.50, A 0.0, B 0.0, C 0.02} –

система начала координат расположена на столе и является основной

системой координат при взятии продуктов из подставки в зоне 1.

BASE[7]={X 581.76, Y -282.56, Z -11.49, A 0.0, B 0.0, C 0.0} – система

начала координат расположена на столе и является основной системой

координат при взятии продуктов из подставки в зоне 1.

Имена для систем координат: TOOL[7]="rs2025"

BASE[6]="rs2025_1"

BASE[7]=" rs2025_2"

*имена даны для удобства, менять их в Р-Про не нужно.

*углы ориентации в системе KSS обозначаются А, В, С: А – поворот вокруг оси z, В – поворот вокруг оси у, С – поворот вокруг оси х.

13. КАРТА СИГНАЛОВ РОБОТА

Выходы робота:

OUT[1] – сигнал на захват (OUT[1] = True – разжать пальцы захвата).

OUT[2] – сигнал на захват (OUT[2] = True – сжать пальцы захвата).

IN[267] – входной сигнал на наличие продуктов в зоне 1.

IN[265] – входной сигнал на наличие продуктов в зоне 2.

* на захват и отпускание объекта требуется время (0.25 сек)



** для разжимания/сжатия пальцев захвата требуется парное использование сигналов:

OUT[1] = True + OUT[2] = False / OUT[1] = False + OUT[2] = True

14. ОБЩИЕ ПРАВИЛА ИМЕНОВАНИЯ В KRL (KUKA ROBOT LANGUAGE)

Максимальная длина имени

В KRL имена (переменных, программ, точек и т.д.) могут быть длиной до 24 символов.

Разрешённые символы

Имя может состоять из букв A–Z, цифр 0–9, символов _.

Имя не может начинаться с цифры.

Не рекомендуется начинать название пользовательских переменных с \$, потому что системные переменные KUKA уже используют этот префикс.

Ключевые слова

Имена не должны совпадать с зарезервированными ключевыми словами KRL (kuka, tool и т.д.).

15. РЕЗУЛЬТАТЫ

Файл.rpro «Модуль_Промробот_Сокращенное Название университета», файл .pdf с инструкцией для оператора «Модуль_Промробот_Сокращенное Название университета», файлы .src и .dat.

Запрещено:

- Использовать заранее подготовленные файлы .rpro.
- Пользоваться заранее написанными скриптами.
- Использовать ИИ.

