Федеральное агентство по образованию

Государственное образовательное учреждение высшего профессионального образования

Московский государственный технический университет им. Н.Э. Баумана

Кафедра «Робототехнические системы и мехатроника»

**Домашнее задание**

По курсу:

**“Устройства управления ПР и РТК”**

Тема:

**«Моделирование и управление ПР с помощью CAM системы RoboDK с использованием языка Python»**

Преподаватель: **к.т.н., доц. Солнцев В.И.**

Студент группы СМ7-12М: Вадим Мовтян

2024

**Цель домашнего задания**

Часть 1. Научиться программировать ПР KUKA KR 4 AGILUS на языке KRL и проводить моделирование движения в среде RoboDK.

Часть 2. Научиться программировать ПР на языке Python.

**Постановка задачи.**

Имеется ПР KUKA KR 4 AGILUS, оснащенный фломастером.

Задание состоит в том, что надо написать программу управления ПР, реализующую рисование фломастером свои имя и фамилию в системе координат BASE на рабочей плоскости.

Домашнее задание состоит из двух частей.

В первой части домашнего задания нужно реализовать возможность изменения местоположения и ориентации записи своих имени и фамилии на рабочей плоскости без использования языка Python.

* Необходимо составить 3D модель ПР в среде RoboDK с держателем фломастера и рабочей зоной.
* Необходимо получить программу управления ПР в среде RoboDK и вывести листинг ее на языке KRL.
* В листинге программы на языке KRL сделать комментарии (комментарии должны быть справа от кодов программы).
* Подготовить программу для реализации в соответствующей лабораторной работе на ПР KUKA KR 4 AGILUS.

Во второй части домашнего задания нужно реализовать возможность изменения местоположения и ориентации записи своих имени и фамилии на рабочей плоскости с использованием языка Python. Предусмотреть возможность масштабирования букв без ручного изменения координат точек с одинаковым масштабом для имени и фамилии. Масштаб букв в имени и фамилии должны вводится масштабным коэффициентом (для 100% - коэффициент 1). Обеспечить перестановку букв в записи своих имени и фамилии местами в заданном порядке для каждого слова. Порядок следования букв в имени и фамилии должны вводится числовой комбинацией с помощью языка Python. При нормальном порядке это комбинация задается цифрами по порядку (12345…). Работоспособность программы должна сохраняться при изменении местоположения слова на рабочей плоскости.

* Необходимо составить 3D модель ПР в среде RoboDK с держателем фломастера и рабочей зоной.
* Необходимо написать программу управления ПР на языке KRL с использованием возможностей ПО RoboDK. Использовать программные вставки на языке Python API.
* Получить текст программы на языке KRL с программными вставками на языке Python API.
* В листинге программы на языке Python и KRL сделать комментарии (комментарии должны быть справа от кодов программы).
* Промоделировать и оптимизировать работу ПР в программном пакете RoboDK. Критерием оптимизации является время выполнения программы.
* Подготовить программу для реализации в соответствующей лабораторной работе на ПР KUKA KR 4 AGILUS.

**Содержание** **домашнего задания**

Смысл задания Frame – чтобы программы рисования свои имя и фамилию не менялись при смещении Frame. Рабочая плоскость совпадает с плоскостью основания робота.

Это касается и программ на Питоне.

В первой части домашнем задании нужно подготовить 2 программы для реализации на ПР KUKA KR 4 (Python не использовать):

**1 программа:**

нарисовать фломастером свои имя и фамилию на рабочей плоскости.

**2 программа:**

нарисовать фломастером свои имя и фамилию сдвинутыми и повернутыми на рабочей плоскости.

Во второй части домашнего задания нужно подготовить 4 программы (использовать Python):

**1 программа:**

нарисовать фломастером свои имя и фамилию на рабочей плоскости.

**2 программа:**

нарисовать фломастером свои имя и фамилию, сдвинутые и повернутые на рабочей плоскости.

**3 программа:**

нарисовать фломастером свои имя и фамилию с изменением масштаба букв без ручного изменения координат точек на рабочей плоскости.

**4 программа:**

нарисовать фломастером свои имя и фамилию с начальными условиями 1 программы с перестановкой букв.

**Порядок выполнения домашнего задания:**

1. Выбор модели манипулятора и вставка инструмента робота

Выбираем манипулятор КUKA KR 4 R600 из библиотеки среды (Рисунок 1.1).

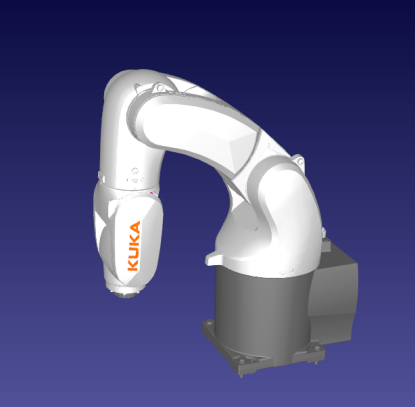


Рис. 1.1. 3D модель КUKA KR 4 R600

В качестве инструмента был выбран держатель с фломастером. (“Держатель фломастера.tool” находится в папке с методичкой). Координаты центра инструмента скорректированы в соответствии с Рисунком 1.2.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рис. 1.2. Координаты центра инструмента по осям

1. Простановка точек для написания букв имени и фамилии

Вначале была задана система координат *Frame1,* которая имеет координаты *(0; 0; 0)* относительно системы координат базы робота. Относительно нее будут расставлены все точки для написания букв.

Было решено записывать каждую букву вместе с отступом после нее в квадраты. Размер квадрата буквы составляет 120x40 мм, отступ – 10 мм.

Ячейки прорисованы оранжевой штрих-пунктирной линией, а оси система координат *Frame1* - черными.

Расположение точек для написания букв изображено на рисунке 2.1.

Изображение выглядит как диаграмма, линия, текст, Параллельный

Автоматически созданное описание

Рис. 2.1. Расположение точек для написания букв.

Помимо точек, образующих сами буквы, имеются еще точки входа и выхода. Это точки, смещенные на 50мм по оси Z вверх относительно *Frame1* и по осям X и Y совпадающие с первой и последней точкой написания буквы соответственно (в подпрограммах имеют название, совпадающее с соответствующей точкой на плоскости с добавлением «\_0»). Из точки входа фломастер сближается с плоскостью перед написанием буквы, а в точку выхода он перемещается по завершению написания. Если первая и последняя точки совпадают (как в букве «В»), то вход и выход представляются одной точкой.

В точку входа и из точки выхода каждой буквы рабочий инструмент перемещается из точки *Workbench* (для имени) или *Workbench\_surname* (для фамилии). Они смещенные на 250мм по оси Z вверх относительно *Frame1*, лежат на оси Y, а по оси Х имеют координаты 500мм и 300мм соответственно {*Workbench* (500; 0; 250), *Workbench\_surname* (300; 0; 250)}.

Весь процесс начинается и заканчивается в начальной точке *home*, заданной относительно системы координат базы робота *home (*537; 0; 720).

Для написания смещенных и развернутых слов была задана система координат *Frame\_turned* (-50; 10; 0) и повернута вокруг оси Z на 60°.

**Часть I**

1. Подпрограммы созданные при помощи графического интерфейса

Поставленная задача решается итерационно. Вначале были составлены подпрограммы для написания буквы, имеющие одинаковую структуру:

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рис 3.1. Структура подпрограммы написания буквы на примере написания «В»

Подпрограммы написания во фрейме состоят из последовательно поставленных программ выбора требуемой системы координат, подпрограмм написания слова и подпрограммы перехода в начальную точку. На рисунке 3.3 а) представлена структура программы написания имени и фамилии в исходном фрейме (Frame1). На рисунке 3.3 б) представлена структура программы написания имени и фамилии в повернутом фрейме (Frame\_turned). Главная подпрограмма состоит из последовательного вызова подпрограмм написания во фрейме (рис 3.3 б).

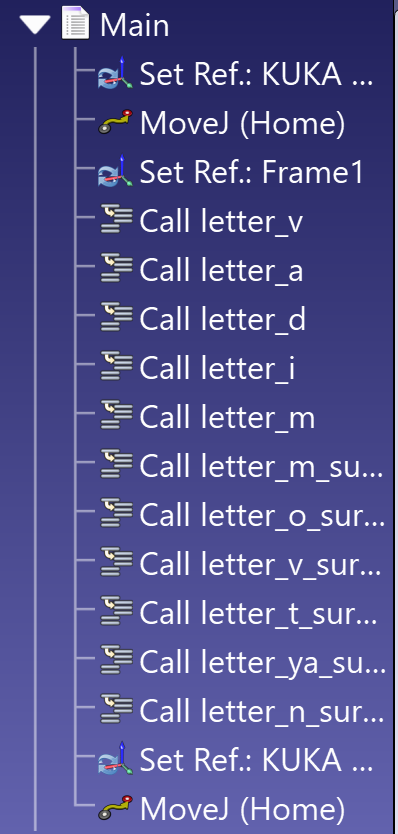
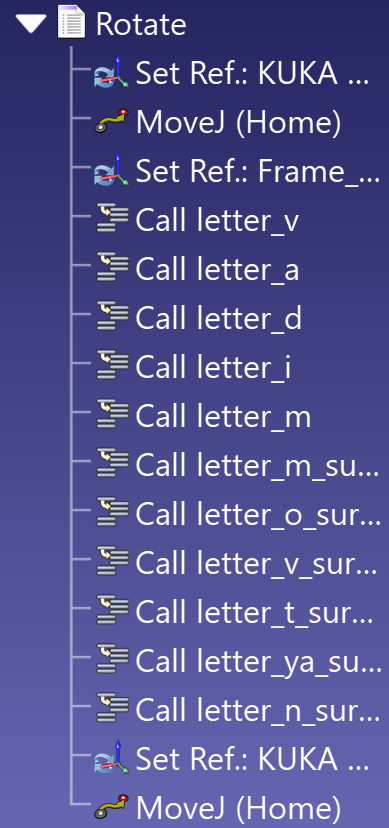
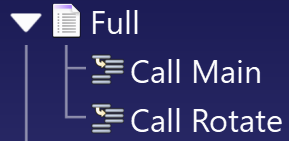
а)  б)  в) 

Рис 3.3. Структура а) подпрограммы написания в исходном фрейме;

б) подпрограммы написания в повернутом фрейме;

в) главной подпрограммы

1. Подпрограммы на языке KRl

При помощи встроенной в RoboDK функции трансляции подпрограмм, созданных при помощи графического интерфейса, в специализированные языки для манипуляторов, были сгенерированы подпрограммы на языке KRL. В силу того, что они дублируют приведенные раннее подпрограммы их описание будет сведено к комментированию кода.

Структура подпрограмм написания буквы на примере написания «А» в имени: **letter\_a**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF Letter\_а ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5 ; Установка количества предсказанных шагов траектории

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:34:17

; Using nominal kinematics.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000} ; Движение PTP в заданную точку

PTP {A1 -18.03430,A2 -35.74850,A3 85.16640,A4 0.00000,A5 40.58210,A6 -18.03430,E1 0.00000} ; Движение PTP в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 630.000,Y 120.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

LIN {X 510.000,Y 100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

LIN {X 546.000,Y 106.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

LIN {X 546.000,Y 106.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 546.000,Y 134.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 546.000,Y 134.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000} ; Движение PTP в заданную точку

END ; Завершение подпрограммы

Структура подпрограмм написания буквы на примере написания «О» в фамилии: **letter\_o\_surname**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF letter\_o\_surname ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5 ; Установка количества предсказанных шагов траектории

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:20

; Using nominal kinematics.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} ; Движение PTP в заданную точку

PTP {A1 -25.01690,A2 -49.95440,A3 115.93100,A4 0.00000,A5 24.02380,A6 -25.01690,E1 0.00000} ; Движение PTP в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 380.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

CIRC {X 360.000,Y 120.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 380.000,Y 100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Круговое движение через промежуточную точку к конечной точке.

LIN {X 460.000,Y 100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

CIRC {X 480.000,Y 120.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 460.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Круговое движение через промежуточную точку к конечной точке.

LIN {X 380.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 380.000,Y 140.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} ; Движение PTP в заданную точку

END ; Завершение подпрограммы

Структура подпрограммы написания имени, фамилии: **Main**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF Main ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:51

; Using nominal kinematics.

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000} ; Выбор базовой системы координат

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} ; Движение PTP в домашнюю точку

letter\_v() ; Вызов подпрограммы

letter\_a() ; Вызов подпрограммы

letter\_d() ; Вызов подпрограммы

letter\_i() ; Вызов подпрограммы

letter\_m() ; Вызов подпрограммы

letter\_m\_surname() ; Вызов подпрограммы

letter\_o\_surname() ; Вызов подпрограммы

letter\_v\_surname() ; Вызов подпрограммы

letter\_t\_surname() ; Вызов подпрограммы

letter\_ya\_surnam() ; Вызов подпрограммы

letter\_n\_surname() ; Вызов подпрограммы

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} ; Движение PTP в домашнюю точку

END ; конец подпрограммы

Структура подпрограммы написания повернутой и смещенной имени, фамилии: **Rotate**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF Rotate ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:36:05

; Using nominal kinematics.

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000} ; Выбор базовой системы координат

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} ; Движение PTP в домашнюю точку

; ---- Setting reference (Base) ----

$BASE = {FRAME: X -50.000,Y 10.000,Z 0.000,A 60.000,B 0.000,C 0.000} ; Выбор смещенной системы координат

; --------------------------

letter\_v() ; Вызов подпрограммы

letter\_a() ; Вызов подпрограммы

letter\_d() ; Вызов подпрограммы

letter\_i() ; Вызов подпрограммы

letter\_m() ; Вызов подпрограммы

letter\_m\_surname() ; Вызов подпрограммы

letter\_o\_surname() ; Вызов подпрограммы

letter\_v\_surname() ; Вызов подпрограммы

letter\_t\_surname() ; Вызов подпрограммы

letter\_ya\_surnam() ; Вызов подпрограммы

letter\_n\_surname() ; Вызов подпрограммы

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000} ; Выбор базовой системы координат

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} ; Движение PTP в домашнюю точку

END ; конец подпрограммы

Главная подпрограмма **Full** состоит из последовательного вызова подпрограмм **Main**() и**Rotate**().

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF Main ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:36:12

; Using nominal kinematics.

Main() ; Вызов подпрограммы

Rotate() ; Вызов подпрограммы

END

Скрипты остальных подпрограмм имеют вид, подобный описанным и приведены в приложении.

На основании полученных скриптов стоит отметить, что в ПО RoboDK нет возможности изменять скорость для перемещения PTP (которая меняется в %), только CP (LIN, CIRC).

**Часть II**

1. Подпрограммы на языке питон

Все подпрограммы имеют одинаковую структуру. Они состоят из групп функций: передвижения, написания букв, написания слов, главных функций.

Вначале прописаны 3 функции для задания правил передвижений: наискорейшего (Joint Move) – move\_J(speed, a, b, c), линейного (Linear Move) - move\_L(speed, a, b, c) и по дуге (CircleMove) – move\_C(speed, (a1, b1, c1), (a2, b2, c2)). На вход они принимают величины смещения по осям X, Y и Z относительно какой-либо заданной точки, а так же заданную скорость передвижения.

Рассмотрим функции написания букв на примере буквы “В” (рис. 3.1). На вход функция принимает k. k - отвечает за масштаб букв.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рис.3.1. Функция написания буквы “В”

Структура функции написания буквы в зависимости от выбранного порядка (рис.3.2. а), б)). На вход подается три аргумента: position, i, scale. Position – буква, которую необходимо нарисовать, i – номер позиции на которой нужно нарисовать выбранную буквы, scale – масштаб.

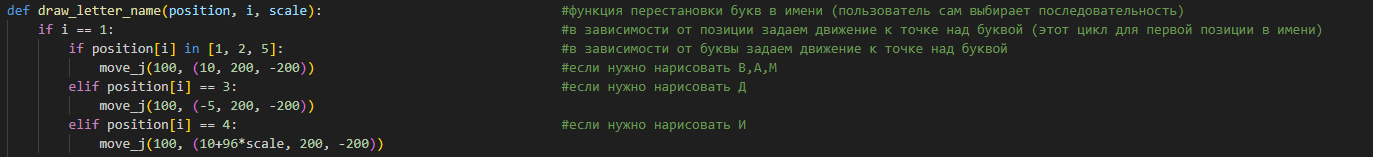


Рис.3.2. а) фрагмент кода, в котором в зависимости от выбранной буквы происходит соответствующее движение к рабочей точке

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рис.3.2. б) фрагмент кода, в котором в зависимости от выбранной буквы происходит отрисовка соответствующей буквы

Ниже на рисунке 3.3. представлен фрагмент кода для отрисовки имени и фамилии. На вход подается аргументы name и surname. Name – 5-ти значное число, которое разделяется в массив поэлементно и передается в функцию draw\_letter\_name в качестве аргумента position. Для surname аналогично.

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рис.3.3. Фрагмент структуры кода для отрисовки имени и фамилии

Последней является главная функция – функция окон для получения масштаба и порядка букв, с последующим вызовом функции для отрисовки имени и фамилии.

1. Подпрограммы на языке KRl

Все подпрограммы имеют одинаковую структуру.

**Name\_surname**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF name\_surname ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 17:04:53

$APO.CPTP = 10.000

$APO.CDIS = 10.000

$VEL.CP = 0.10000

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000} ; Установка базовой системы координат.

; --------------------------

; ---- Setting tool (TCP) ----

$TOOL = {FRAME: X -80.000,Y 0.000,Z 152.000,A 180.000,B 0.000,C 180.000} ; Установка параметров TCP

; --------------------------

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 -24.94390,A2 -32.55110,A3 78.53440,A4 0.00000,A5 44.01670,A6 -24.94390,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 606.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

CIRC {X 586.000,Y 184.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 566.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Круговое движение через промежуточную точку к конечной точке.

CIRC {X 538.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Круговое движение через промежуточную точку к конечной точке.

END ; конец подпрограммы

; Последующие команды аналогичны: Установка скорости и выполнение PTP или LIN движений.

**turn**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF turn ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 17:05:25

$APO.CPTP = 10.000

$APO.CDIS = 10.000

$VEL.CP = 0.10000

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 10.000,Y 0.000,Z 0.000,A 60.000,B 0.000,C 0.000} ; Установка повернутой системы координат.

; --------------------------

; ---- Setting tool (TCP) ----

$TOOL = {FRAME: X -80.000,Y 0.000,Z 152.000,A 180.000,B 0.000,C 180.000} ; Установка параметров TCP

; --------------------------

PTP {A1 -58.83260,A2 -62.35990,A3 86.77100,A4 0.00000,A5 65.58890,A6 1.16736,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 -83.74280,A2 -32.40160,A3 78.22590,A4 0.00000,A5 44.17570,A6 -23.74280,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 606.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

CIRC {X 586.000,Y 184.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 566.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Круговое движение через промежуточную точку к конечной точке.

CIRC {X 538.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Круговое движение через промежуточную точку к конечной точке.

; Последующие команды аналогичны: ; Установка скорости и выполнение PTP или LIN движений.

**scale**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF scale ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 17:22:49

$APO.CPTP = 10.000

$APO.CDIS = 10.000

$VEL.CP = 0.10000

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 120.000,B 0.000,C 0.000} ; Установка базовой системы координат.

; --------------------------

; ---- Setting tool (TCP) ----

$TOOL = {FRAME: X -80.000,Y 0.000,Z 152.000,A 180.000,B 0.000,C 180.000} ; Установка параметров TCP

; --------------------------

PTP {A1 -120.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 -144.94400,A2 -32.55110,A3 78.53440,A4 0.00000,A5 44.01670,A6 -24.94390,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 586.800,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

; Последующие команды аналогичны: ; Установка скорости и выполнение PTP или LIN движений.

**order**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF order ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 17:08:24

$APO.CPTP = 10.000

$APO.CDIS = 10.000

$VEL.CP = 0.10000

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A -60.000,B 0.000,C 0.000} ; Установка базовой системы координат.

; --------------------------

; ---- Setting tool (TCP) ----

$TOOL = {FRAME: X -80.000,Y 0.000,Z 152.000,A 180.000,B 0.000,C 180.000} ; Установка параметров TCP

; --------------------------

PTP {A1 60.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 35.05610,A2 -32.55110,A3 78.53440,A4 0.00000,A5 44.01670,A6 -24.94390,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 606.000,Y 192.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

; Последующие команды аналогичны: ; Установка скорости и выполнение PTP или LIN движений.

На месте обрыва в коде идет продолжение когда, состоящего из:

* $VEL.CP = 0.50000 - установки скорости для движения CP
* PTP{A1 \_,A2 \_,A3 \_,A4 \_,A5 \_,A6 \_,E1 0.00000}, LIN{X \_, Y \_, Z \_, A \_, B \_, C \_}, CIRC{X \_, Y \_, Z \_, A \_, B \_, C \_},{X \_, Y \_, Z \_, A \_, B \_, C \_} - PTP перемещения, линейного перемещения или перемещения по окружности (на месте нижнего подчеркивания находятся системы координат).

**Выводы:**

В ПО RoboDK нет возможности изменять скорость для перемещения PTP.

ПО RoboDK является хорошим инструментом для создания программ для управления манипуляторами, способный транслировать подпрограммы на языке питон на язык для робота. Однако для серьезной работы стоит работать непосредственно на языке, разработанном компанией-изготовителем.

Результатом лабораторной работы является создание и отладка программ управления ПР KUKA KR 4 AGILUS в среде RoboDK с использованием дополнительных возможностей программирования ПР KUKA на языке KRL.

# Приложение 1 (KRL)

**Full**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF Full ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:36:12

; Using nominal kinematics.

name\_surname()

name\_surname\_tur()

END

**Main**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF Main ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:51

; Using nominal kinematics.

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000} ; Выбор базовой системы координат

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} ; Движение PTP в домашнюю точку

letter\_v() ; Вызов подпрограммы

letter\_a() ; Вызов подпрограммы

letter\_d() ; Вызов подпрограммы

letter\_i() ; Вызов подпрограммы

letter\_m() ; Вызов подпрограммы

letter\_m\_surname() ; Вызов подпрограммы

letter\_o\_surname() ; Вызов подпрограммы

letter\_v\_surname() ; Вызов подпрограммы

letter\_t\_surname() ; Вызов подпрограммы

letter\_ya\_surnam() ; Вызов подпрограммы

letter\_n\_surname() ; Вызов подпрограммы

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} ; Движение PTP в домашнюю точку

END ; конец подпрограммы

**Rotate**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF Rotate ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:36:05

; Using nominal kinematics.

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000} ; Выбор базовой системы координат

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} ; Движение PTP в домашнюю точку

; ---- Setting reference (Base) ----

$BASE = {FRAME: X -50.000,Y 10.000,Z 0.000,A 60.000,B 0.000,C 0.000} ; Выбор смещенной системы координат

; --------------------------

letter\_v() ; Вызов подпрограммы

letter\_a() ; Вызов подпрограммы

letter\_d() ; Вызов подпрограммы

letter\_i() ; Вызов подпрограммы

letter\_m() ; Вызов подпрограммы

letter\_m\_surname() ; Вызов подпрограммы

letter\_o\_surname() ; Вызов подпрограммы

letter\_v\_surname() ; Вызов подпрограммы

letter\_t\_surname() ; Вызов подпрограммы

letter\_ya\_surnam() ; Вызов подпрограммы

letter\_n\_surname() ; Вызов подпрограммы

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000} ; Выбор базовой системы координат

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} ; Движение PTP в домашнюю точку

END ; конец подпрограммы

**letter\_a**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF Letter\_а ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5 ; Установка количества предсказанных шагов траектории

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:34:17

; Using nominal kinematics.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000} ; Движение PTP в заданную точку

PTP {A1 -18.03430,A2 -35.74850,A3 85.16640,A4 0.00000,A5 40.58210,A6 -18.03430,E1 0.00000} ; Движение PTP в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 630.000,Y 120.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

LIN {X 510.000,Y 100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

LIN {X 546.000,Y 106.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

LIN {X 546.000,Y 106.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 546.000,Y 134.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 546.000,Y 134.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000} ; Движение PTP в заданную точку

END ; Завершение подпрограммы

**letter\_o\_surname**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF letter\_o\_surname ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5 ; Установка количества предсказанных шагов траектории

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:20

; Using nominal kinematics.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} ; Движение PTP в заданную точку

PTP {A1 -25.01690,A2 -49.95440,A3 115.93100,A4 0.00000,A5 24.02380,A6 -25.01690,E1 0.00000} ; Движение PTP в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 380.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

CIRC {X 360.000,Y 120.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 380.000,Y 100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Круговое движение через промежуточную точку к конечной точке.

LIN {X 460.000,Y 100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

CIRC {X 480.000,Y 120.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 460.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Круговое движение через промежуточную точку к конечной точке.

LIN {X 380.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 380.000,Y 140.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} ; Линейное движение в заданную точку

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} ; Движение PTP в заданную точку

END ; Завершение подпрограммы

**letter\_v**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_v ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:37:19

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000}

PTP {A1 -24.94390,A2 -32.55110,A3 78.53440,A4 0.00000,A5 44.01670,A6 -24.94390,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 630.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

CIRC {X 605.000,Y 180.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 580.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

CIRC {X 545.000,Y 160.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 510.000,Y 200.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000}

END

**letter\_d**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_d ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:34:29

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000}

PTP {A1 -10.91110,A2 -39.67910,A3 93.42620,A4 0.00000,A5 36.25300,A6 -10.91110,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 495.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 510.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 510.000,Y 40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 495.000,Y 40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 495.000,Y 40.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 510.000,Y 75.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 510.000,Y 75.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 630.000,Y 75.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 630.000,Y 45.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 510.000,Y 45.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 510.000,Y 45.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000}

END

**letter\_i**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_i ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:34:52

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000}

PTP {A1 -2.08257,A2 -18.34680,A3 49.71360,A4 0.00000,A5 58.63320,A6 -2.08257,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 630.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 510.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 630.000,Y -20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 510.000,Y -20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 510.000,Y -20.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000}

END

**letter\_m**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_m ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:02

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000}

PTP {A1 5.31455,A2 -38.49360,A3 90.92080,A4 0.00000,A5 37.57280,A6 5.31455,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 510.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 630.000,Y -50.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 510.000,Y -60.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 630.000,Y -70.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 510.000,Y -80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 510.000,Y -80.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 0.00000,E1 0.00000}

END

**letter\_m\_surname**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_m\_surname ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:09

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

PTP {A1 -35.53770,A2 -48.63810,A3 112.93800,A4 0.00000,A5 25.70030,A6 -35.53770,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 480.000,Y 190.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 360.000,Y 180.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 480.000,Y 170.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 360.000,Y 160.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y 160.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

END

**letter\_v\_surname**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_v\_surname ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:26

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

PTP {A1 -15.94540,A2 -53.70260,A3 124.76400,A4 0.00000,A5 18.93900,A6 -15.94540,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 480.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

CIRC {X 455.000,Y 60.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 430.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

CIRC {X 395.000,Y 40.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 360.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y 80.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

END

**letter\_t\_surname**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_t\_surname ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:32

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

PTP {A1 -2.86241,A2 -42.39960,A3 99.23040,A4 0.00000,A5 33.16920,A6 -2.86241,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 480.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 480.000,Y -20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 480.000,Y -20.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 480.000,Y 0.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 480.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 360.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y 0.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

END

**letter\_ya\_surname**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_ya\_surnam ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:38

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

PTP {A1 8.13010,A2 -54.43520,A3 126.56300,A4 0.00000,A5 17.87210,A6 8.13010,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 420.000,Y -80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

CIRC {X 450.000,Y -50.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 480.000,Y -80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 360.000,Y -80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y -80.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

END

**letter\_n\_surname**

&ACCESS RVP

&REL 1

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe

&PARAM EDITMASK = \*

DEF letter\_n\_surname ( )

;FOLD Initialise and set default speed

BAS (#INITMOV,0)

BAS (#VEL\_PTP,100)

BAS (#ACC\_PTP,20)

$VEL.CP=0.2

BAS (#TOOL,0)

BAS (#BASE,0)

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 12:35:44

; Using nominal kinematics.

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

PTP {A1 19.65380,A2 -53.15460,A3 123.43600,A4 0.00000,A5 19.71840,A6 19.65380,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 480.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 480.000,Y -100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 480.000,Y -140.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 480.000,Y -140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 360.000,Y -140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 360.000,Y -140.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 420.000,Y -100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

LIN {X 420.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.07500

LIN {X 420.000,Y -140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.05000

LIN {X 420.000,Y -140.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000}

$VEL.CP = 0.10000

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000}

END

# Приложение 2 (KRL)

**Name\_surname**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF name\_surname ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 17:04:53

$APO.CPTP = 10.000

$APO.CDIS = 10.000

$VEL.CP = 0.10000

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000} ; Установка базовой системы координат.

; --------------------------

; ---- Setting tool (TCP) ----

$TOOL = {FRAME: X -80.000,Y 0.000,Z 152.000,A 180.000,B 0.000,C 180.000} ; Установка параметров TCP

; --------------------------

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 -24.94390,A2 -32.55110,A3 78.53440,A4 0.00000,A5 44.01670,A6 -24.94390,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 606.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

CIRC {X 586.000,Y 184.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 566.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Круговое движение через промежуточную точку к конечной точке.

CIRC {X 538.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Круговое движение через промежуточную точку к конечной точке.

; Последующие команды аналогичны: ; Установка скорости и выполнение PTP или LIN движений.

$VEL.CP = 0.05000

LIN {X 510.000,Y 200.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -18.03430,A2 -35.74850,A3 85.16640,A4 0.00000,A5 40.58210,A6 -18.03430,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 510.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y 124.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y 108.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y 114.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y 114.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 546.000,Y 134.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y 134.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -10.91110,A2 -39.67910,A3 93.42620,A4 0.00000,A5 36.25300,A6 -10.91110,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 495.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 495.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 495.000,Y 48.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 76.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 76.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 603.000,Y 76.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 603.000,Y 52.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 52.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 52.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -2.17750,A2 -23.59020,A3 60.25220,A4 0.00000,A5 53.33800,A6 -2.17750,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 606.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y -12.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -12.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y -12.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 5.31455,A2 -38.49360,A3 90.92080,A4 0.00000,A5 37.57280,A6 5.31455,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 510.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y -48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -56.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y -64.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y -72.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -35.53770,A2 -48.63810,A3 112.93800,A4 0.00000,A5 25.70030,A6 -35.53770,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 192.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 184.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 176.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 168.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -25.31290,A2 -50.31200,A3 116.75200,A4 0.00000,A5 23.56000,A6 -25.31290,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 376.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 360.000,Y 124.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 376.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 440.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 456.000,Y 124.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 440.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 376.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 376.000,Y 140.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -15.94540,A2 -53.70260,A3 124.76400,A4 0.00000,A5 18.93900,A6 -15.94540,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 436.000,Y 64.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 416.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 388.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 360.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 80.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -3.04478,A2 -45.16320,A3 105.22200,A4 0.00000,A5 29.94130,A6 -3.04478,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 456.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y -12.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y -12.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 4.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 4.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 4.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 4.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 8.13010,A2 -54.43520,A3 126.56300,A4 0.00000,A5 17.87210,A6 8.13010,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 408.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 432.000,Y -48.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 456.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y -72.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 19.65380,A2 -53.15460,A3 123.43600,A4 0.00000,A5 19.71840,A6 19.65380,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y -100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y -132.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y -132.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y -132.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y -132.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y -100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 408.000,Y -132.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y -132.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 0.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} C\_PTP

END

**turn**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF turn ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 17:05:25

$APO.CPTP = 10.000

$APO.CDIS = 10.000

$VEL.CP = 0.10000

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 10.000,Y 0.000,Z 0.000,A 60.000,B 0.000,C 0.000} ; Установка повернутой системы координат.

; --------------------------

; ---- Setting tool (TCP) ----

$TOOL = {FRAME: X -80.000,Y 0.000,Z 152.000,A 180.000,B 0.000,C 180.000} ; Установка параметров TCP

; --------------------------

PTP {A1 -58.83260,A2 -62.35990,A3 86.77100,A4 0.00000,A5 65.58890,A6 1.16736,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 -83.74280,A2 -32.40160,A3 78.22590,A4 0.00000,A5 44.17570,A6 -23.74280,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 606.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

CIRC {X 586.000,Y 184.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 566.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Круговое движение через промежуточную точку к конечной точке.

CIRC {X 538.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Круговое движение через промежуточную точку к конечной точке.

; Последующие команды аналогичны: ; Установка скорости и выполнение PTP или LIN движений.

$VEL.CP = 0.05000

LIN {X 510.000,Y 200.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -58.83260,A2 -62.35990,A3 86.77100,A4 0.00000,A5 65.58890,A6 1.16736,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -76.80070,A2 -35.44350,A3 84.53080,A4 0.00000,A5 40.91280,A6 -16.80070,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 510.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y 124.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y 108.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y 114.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y 114.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 546.000,Y 134.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y 134.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -58.83260,A2 -62.35990,A3 86.77100,A4 0.00000,A5 65.58890,A6 1.16736,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -69.64010,A2 -39.24880,A3 92.51520,A4 0.00000,A5 36.73360,A6 -9.64006,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 495.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 495.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 495.000,Y 48.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 76.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 76.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 603.000,Y 76.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 603.000,Y 52.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 52.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 52.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -58.83260,A2 -62.35990,A3 86.77100,A4 0.00000,A5 65.58890,A6 1.16736,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -61.22340,A2 -22.63250,A3 58.31970,A4 0.00000,A5 54.31280,A6 -1.22339,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 606.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y -12.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -12.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y -12.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -58.83260,A2 -62.35990,A3 86.77100,A4 0.00000,A5 65.58890,A6 1.16736,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -53.61730,A2 -37.72250,A3 89.29810,A4 0.00000,A5 38.42450,A6 6.38272,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 510.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y -48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -56.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y -64.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y -72.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -58.83260,A2 -62.35990,A3 86.77100,A4 0.00000,A5 65.58890,A6 1.16736,E1 0.00000} C\_PTP

PTP {A1 -57.79580,A2 -100.54900,A3 130.63600,A4 0.00000,A5 59.91380,A6 2.20423,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -93.87610,A2 -48.72230,A3 113.12800,A4 0.00000,A5 25.59440,A6 -33.87610,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 192.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 184.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 176.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 168.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -57.79580,A2 -100.54900,A3 130.63600,A4 0.00000,A5 59.91380,A6 2.20423,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -83.57380,A2 -50.21670,A3 116.53300,A4 0.00000,A5 23.68400,A6 -23.57380,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 376.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 360.000,Y 124.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 376.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 440.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 456.000,Y 124.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 440.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 376.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 376.000,Y 140.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -57.79580,A2 -100.54900,A3 130.63600,A4 0.00000,A5 59.91380,A6 2.20423,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -74.05320,A2 -53.47190,A3 124.20300,A4 0.00000,A5 19.26890,A6 -14.05320,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 436.000,Y 64.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 416.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 388.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 360.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 80.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -57.79580,A2 -100.54900,A3 130.63600,A4 0.00000,A5 59.91380,A6 2.20423,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -61.70480,A2 -44.64220,A3 104.08400,A4 0.00000,A5 30.55840,A6 -1.70480,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 456.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y -12.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y -12.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 4.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 4.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 4.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 4.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -57.79580,A2 -100.54900,A3 130.63600,A4 0.00000,A5 59.91380,A6 2.20423,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -50.31090,A2 -53.88660,A3 125.21300,A4 0.00000,A5 18.67370,A6 9.68912,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 408.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 432.000,Y -48.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 456.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y -72.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -57.79580,A2 -100.54900,A3 130.63600,A4 0.00000,A5 59.91380,A6 2.20423,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -39.12990,A2 -52.45140,A3 121.75400,A4 0.00000,A5 20.69790,A6 20.87010,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y -100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y -132.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y -132.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y -132.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y -132.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y -100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 408.000,Y -132.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y -132.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -57.79580,A2 -100.54900,A3 130.63600,A4 0.00000,A5 59.91380,A6 2.20423,E1 0.00000} C\_PTP

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} C\_PTP

END

**scale**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF scale ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 17:22:49

$APO.CPTP = 10.000

$APO.CDIS = 10.000

$VEL.CP = 0.10000

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 120.000,B 0.000,C 0.000} ; Установка базовой системы координат.

; --------------------------

; ---- Setting tool (TCP) ----

$TOOL = {FRAME: X -80.000,Y 0.000,Z 152.000,A 180.000,B 0.000,C 180.000} ; Установка параметров TCP

; --------------------------

PTP {A1 -120.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 -144.94400,A2 -32.55110,A3 78.53440,A4 0.00000,A5 44.01670,A6 -24.94390,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 586.800,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

; Последующие команды аналогичны: ; Установка скорости и выполнение PTP или LIN движений.

$VEL.CP = 0.07500

CIRC {X 570.800,Y 187.200,Z 0.000,A 0.000,B 0.000,C 0.000},{X 554.800,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 532.400,Y 174.400,Z 0.000,A 0.000,B 0.000,C 0.000},{X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y 200.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -138.03400,A2 -35.74850,A3 85.16640,A4 0.00000,A5 40.58210,A6 -18.03430,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 510.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 586.800,Y 127.200,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y 114.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y 114.400,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 538.800,Y 119.200,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 538.800,Y 119.200,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 538.800,Y 135.200,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 538.800,Y 135.200,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -130.91100,A2 -39.67910,A3 93.42620,A4 0.00000,A5 36.25300,A6 -10.91110,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 495.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 504.600,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 504.600,Y 54.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 495.000,Y 54.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 495.000,Y 54.400,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 504.600,Y 76.800,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 504.600,Y 76.800,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 581.400,Y 76.800,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 581.400,Y 57.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 504.600,Y 57.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 504.600,Y 57.600,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -122.26000,A2 -27.17720,A3 67.52420,A4 0.00000,A5 49.65300,A6 -2.25991,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 586.800,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 586.800,Y -5.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -5.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y -5.600,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -114.68500,A2 -38.49360,A3 90.92080,A4 0.00000,A5 37.57280,A6 5.31455,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 510.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 586.800,Y -46.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -52.800,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 586.800,Y -59.200,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -65.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y -65.600,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

PTP {A1 -120.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -155.53800,A2 -48.63810,A3 112.93800,A4 0.00000,A5 25.70030,A6 -35.53770,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 436.800,Y 193.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 187.200,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 436.800,Y 180.800,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 174.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 174.400,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -145.55400,A2 -50.59510,A3 117.40500,A4 0.00000,A5 23.19030,A6 -25.55440,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 372.800,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 360.000,Y 127.200,Z 0.000,A 0.000,B 0.000,C 0.000},{X 372.800,Y 114.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 424.000,Y 114.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 436.800,Y 127.200,Z 0.000,A 0.000,B 0.000,C 0.000},{X 424.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 372.800,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 372.800,Y 140.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -135.94500,A2 -53.70260,A3 124.76400,A4 0.00000,A5 18.93900,A6 -15.94540,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 436.800,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 420.800,Y 67.200,Z 0.000,A 0.000,B 0.000,C 0.000},{X 404.800,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 382.400,Y 54.400,Z 0.000,A 0.000,B 0.000,C 0.000},{X 360.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 80.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -123.20800,A2 -47.25200,A3 109.83100,A4 0.00000,A5 27.42090,A6 -3.20829,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 436.800,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 436.800,Y -5.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 436.800,Y -5.600,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 436.800,Y 10.400,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 436.800,Y 10.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 10.400,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 10.400,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -111.87000,A2 -54.43520,A3 126.56300,A4 0.00000,A5 17.87210,A6 8.13010,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 398.400,Y -65.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 417.600,Y -46.400,Z 0.000,A 0.000,B 0.000,C 0.000},{X 436.800,Y -65.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y -65.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y -65.600,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 -100.34600,A2 -53.15460,A3 123.43600,A4 0.00000,A5 19.71840,A6 19.65380,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 436.800,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 436.800,Y -100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 436.800,Y -125.600,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 436.800,Y -125.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y -125.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y -125.600,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 398.400,Y -100.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 398.400,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 398.400,Y -125.600,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 398.400,Y -125.600,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 -120.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 -0.00000,E1 0.00000} C\_PTP

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} C\_PTP

END

**Order**

&ACCESS RVP ; Устанавливается права доступа к файлу (RVP - стандартный уровень прав)

&REL 1 ; Версия или ревизия файла

&PARAM TEMPLATE = C:\KRC\Roboter\Template\vorgabe ; Путь к шаблону программы

&PARAM EDITMASK = \* ; Разрешение на полное редактирование файла

DEF order ( ) ; Объявление/начало подпрограммы

;FOLD Initialise and set default speed

BAS (#INITMOV,0) ; Инициализация стандартных параметров движения

BAS (#VEL\_PTP,100) ; Установка скорости PTP движения на 100%

BAS (#ACC\_PTP,20) ; Установка ускорения для PTP на 20%

$VEL.CP=0.2 ; Установка скорости для линейного движения (CP) на 0.2 м/с

BAS (#TOOL,0) ; Установка инструмента №0

BAS (#BASE,0) ; Установка базовой системы координат №0

;ENDFOLD

$ADVANCE = 5

; Program generated by RoboDK v5.2.5 for KUKA KR 4 R600 on 07/01/2025 17:08:24

$APO.CPTP = 10.000

$APO.CDIS = 10.000

$VEL.CP = 0.10000

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A -60.000,B 0.000,C 0.000} ; Установка базовой системы координат.

; --------------------------

; ---- Setting tool (TCP) ----

$TOOL = {FRAME: X -80.000,Y 0.000,Z 152.000,A 180.000,B 0.000,C 180.000} ; Установка параметров TCP

; --------------------------

PTP {A1 60.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.10000 ; Установка скорости для линейного движения (CP) на 0.1 м/с

PTP {A1 35.05610,A2 -32.55110,A3 78.53440,A4 0.00000,A5 44.01670,A6 -24.94390,E1 0.00000} C\_PTP ; Движение в заданную PTP-позицию.

$VEL.CP = 0.05000 ; Установка скорости для линейного движения (CP) на 0.05 м/с

LIN {X 510.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

$VEL.CP = 0.07500 ; Установка скорости для линейного движения (CP) на 0.075 м/с

LIN {X 606.000,Y 192.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS ; Линейное движение в заданную точку

; Последующие команды аналогичны: ; Установка скорости и выполнение PTP или LIN движений.

$VEL.CP = 0.07500

LIN {X 510.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y 168.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 45.09570,A2 -19.77440,A3 52.57320,A4 0.00000,A5 57.20130,A6 -14.90430,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 606.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y 108.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 49.08890,A2 -39.67910,A3 93.42620,A4 0.00000,A5 36.25300,A6 -10.91110,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 495.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 495.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 495.000,Y 48.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 76.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 76.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 603.000,Y 76.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 603.000,Y 52.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 507.000,Y 52.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 507.000,Y 52.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 57.33700,A2 -38.67480,A3 91.30280,A4 0.00000,A5 37.37200,A6 -2.66300,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 510.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y 4.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 510.000,Y -12.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y -12.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y -6.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y -6.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 546.000,Y 14.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 546.000,Y 14.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 65.31450,A2 -38.49360,A3 90.92080,A4 0.00000,A5 37.57280,A6 5.31455,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 510.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 606.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 586.000,Y -56.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 566.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 538.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 510.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 510.000,Y -40.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -63.30450,A3 88.14660,A4 0.00000,A5 65.15780,A6 -0.00000,E1 0.00000} C\_PTP

PTP {A1 60.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 24.46230,A2 -48.63810,A3 112.93800,A4 0.00000,A5 25.70030,A6 -35.53770,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 200.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 168.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 168.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y 200.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y 200.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 408.000,Y 168.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 408.000,Y 168.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 33.43490,A2 -51.69980,A3 119.97800,A4 0.00000,A5 21.72220,A6 -26.56510,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 140.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 408.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 432.000,Y 132.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 456.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 108.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 47.98850,A2 -44.27350,A3 103.28100,A4 0.00000,A5 30.99250,A6 -12.01150,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 456.000,Y 80.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 48.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 48.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 64.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 456.000,Y 64.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y 64.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 64.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 55.91440,A2 -54.61860,A3 127.01900,A4 0.00000,A5 17.60000,A6 -4.08562,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 436.000,Y 4.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 416.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 388.000,Y -12.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 360.000,Y 20.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y 20.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 67.69610,A2 -53.03060,A3 123.13800,A4 0.00000,A5 19.89280,A6 7.69605,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 376.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 360.000,Y -56.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 376.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 440.000,Y -72.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

CIRC {X 456.000,Y -56.000,Z 0.000,A 0.000,B 0.000,C 0.000},{X 440.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 376.000,Y -40.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 376.000,Y -40.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

$VEL.CP = 0.10000

PTP {A1 79.65380,A2 -53.15460,A3 123.43600,A4 0.00000,A5 19.71840,A6 19.65380,E1 0.00000} C\_PTP

$VEL.CP = 0.05000

LIN {X 360.000,Y -100.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y -108.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y -116.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 456.000,Y -124.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.07500

LIN {X 360.000,Y -132.000,Z 0.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

$VEL.CP = 0.05000

LIN {X 360.000,Y -132.000,Z 50.000,A 0.000,B 0.000,C 0.000,E1 0.00000} C\_DIS

PTP {A1 60.00000,A2 -101.70200,A3 131.55600,A4 0.00000,A5 60.14530,A6 0.00000,E1 0.00000} C\_PTP

; ---- Setting reference (Base) ----

$BASE = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; BASE\_DATA[1] = {FRAME: X 0.000,Y 0.000,Z 0.000,A 0.000,B 0.000,C 0.000}

; $BASE = BASE\_DATA[1]

; --------------------------

PTP {A1 0.00000,A2 -90.00000,A3 90.00000,A4 0.00000,A5 0.00000,A6 0.00000,E1 0.00000} C\_PTP

END

# Приложение 3 (Python)

**Name\_surname**

from robolink import \*                                                  # RoboDK API

from robodk import \*                                                    # Robot toolbox

import tkinter as tk

from tkinter import \*

from tkinter import messagebox

#import numpy as np

RDK = Robolink()

robot = RDK.ItemUserPick('Select a robot', ITEM\_TYPE\_ROBOT)             #инициализация робота

if not robot.Valid():

    raise Exception('No robot selected or available')

robot.setPoseFrame(robot.PoseFrame())

robot.setPoseTool(robot.PoseTool())

robot.setRounding(10)

robot.setSpeed(100)

base\_frame = RDK.Item('KUKA KR 4 R600 Base')

Base = RDK.Item('Base')                                                 #определение домашнего положения

Workbench = RDK.Item('Workbench1')

Workbench\_surname = RDK.Item('Workbench\_surname1')

def move\_l(speed, offset):                                              #функция линейного движения

    robot.setSpeed(speed)                                               #задание скорости движения робота

    point = robot.Pose()\*transl(offset[0],offset[1],offset[2])          #расчет желаемой координаты

    robot.MoveL(point)                                                  #линейное движение к определённой координате

def move\_c(speed, offset\_1, offset\_2):                                  #функция дугообразного движения

    robot.setSpeed(speed)                                               #задание скорости движения робота

    point1 = robot.Pose()\*transl(offset\_1[0],offset\_1[1],offset\_1[2])   #промежуточная координата

    point2 = robot.Pose()\*transl(offset\_2[0],offset\_2[1],offset\_2[2])   #конечная координата

    robot.MoveC(point1,point2)                                          #дугообразное движение к определённой координате

def move\_j(speed, offset):                                              #Функция ptp движения

    robot.setSpeed(speed)                                               #Задание скорости движения робота

    point = robot.Pose()\*transl(offset[0],offset[1],offset[2])          #расчет желаемой координаты

    robot.MoveJ(point)                                                  #ptp движение к определённой координате

def letter\_v(k):                                                        #функция написания буквы В

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (96\*k, 0, 0))                                           #линейное движение к В1

    move\_c(75,(-20\*k, -16\*k, 0) , (-40\*k, 0, 0))                        #circle движение В2 В3

    move\_c(75,(-28\*k, -32\*k, 0) , (-56\*k, 0, 0))                        #circle движение В4 В5

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

def letter\_a(k):                                                        #функция написания буквы А

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (96\*k, -16\*k, 0))                                       #линейные движения

    move\_l(75, (-96\*k, -16\*k, 0))

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

    move\_l(50, (36\*k, 6\*k, 0))

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (0, 20\*k, 0))

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

def letter\_d(k):                                                        #функция написания буквы Д

    move\_l(50, (0, 0, -50))

    move\_l(75, (12\*k, 0, 0))

    move\_l(75, (0, -32\*k, 0))

    move\_l(75, (-12\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (12\*k, 28\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, 0, 0))

    move\_l(75, (0, -24\*k, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_i(k):                                                        #функция написания буквы И

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(75, (96\*k, -32\*k, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_m(k):                                                        #функция написания буквы М

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, -8\*k, 0))

    move\_l(75, (-96\*k, -8\*k, 0))

    move\_l(75, (96\*k, -8\*k, 0))

    move\_l(75, (-96\*k, -8\*k, 0))

    move\_l(50, (0, 0, 50))

def letter\_o(k):                                                        #функция написания буквы О

    move\_l(50, (0, 0, -50))

    move\_c(75,(-16\*k, -16\*k, 0) , (0, -32\*k, 0))

    move\_l(75, (64\*k, 0, 0))

    move\_c(75,(16\*k, 16\*k, 0) , (0, 32\*k, 0))

    move\_l(75, (-64\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_t(k):                                                        #функция написания буквы Т

    move\_l(50, (0, 0, -50))

    move\_l(75, (0, -32\*k, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (0, 16, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_ya(k):                                                        #функция написания буквы Я

    move\_l(50, (0, 0, -50))

    move\_l(75, (48\*k, -32\*k, 0))

    move\_c(75,(24\*k, 24\*k, 0) , (48\*k, 0, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_n(k):                                                        #функция написания буквы Н

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (0, -32\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (48\*k, 32\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (0, -32\*k, 0))

    move\_l(50, (0, 0, 50))

def name\_surname():

    robot.setPoseFrame(RDK.Item('Frame1'))

    scale=1

    robot.MoveJ(Workbench.Pose())                                       #движение в точку над Именем

    move\_j(100, (10, 200, -200))                                        #движение  точку над буквой

    letter\_v(scale)                                                     #отрисовка буквы В

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10, 140, -200))

    letter\_a(scale)                                                     #отрисовка буквы А

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (-5, 80, -200))

    letter\_d(scale)                                                     #отрисовка буквы Д

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10+96\*scale, 20, -200))

    letter\_i(scale)                                                     #отрисовка буквы И

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10, -40, -200))

    letter\_m(scale)                                                     #отрисовка буквы М

    robot.MoveJ(Workbench.Pose())

    robot.MoveJ(Workbench\_surname.Pose())                               #движение в точку над Фамилией

    move\_j(100, (60, 200, -200))                                        #движение  точку над буквой

    letter\_m(scale)                                                     #отрисовка буквы М

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60+16\*scale, 140, -200))

    letter\_o(scale)                                                     #отрисовка буквы О

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, 80, -200))

    letter\_v(scale)                                                     #отрисовка буквы В

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60+96\*scale, 20, -200))

    letter\_t(scale)                                                     #отрисовка буквы Т

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, -40, -200))

    letter\_ya(scale)                                                    #отрисовка буквы Я

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, -100, -200))

    letter\_n(scale)                                                     #отрисовка буквы Н

    robot.MoveJ(Workbench\_surname.Pose())

    robot.setPoseFrame(base\_frame)

    robot.MoveJ(Base.Pose())                                            #возврат в исходное положение

name\_surname()

**turn**

from robolink import \*                                                  # RoboDK API

from robodk import \*                                                    # Robot toolbox

import tkinter as tk

from tkinter import \*

from tkinter import messagebox

#import numpy as np

RDK = Robolink()

robot = RDK.ItemUserPick('Select a robot', ITEM\_TYPE\_ROBOT)             #инициализация робота

if not robot.Valid():

    raise Exception('No robot selected or available')

robot.setPoseFrame(robot.PoseFrame())

robot.setPoseTool(robot.PoseTool())

robot.setRounding(10)

robot.setSpeed(100)

base\_frame = RDK.Item('KUKA KR 4 R600 Base')

Base = RDK.Item('Base')                                                 #определение домашнего положения

Workbench = RDK.Item('Workbench2')

Workbench\_surname = RDK.Item('Workbench\_surname2')

def move\_l(speed, offset):                                              #функция линейного движения

    robot.setSpeed(speed)                                               #задание скорости движения робота

    point = robot.Pose()\*transl(offset[0],offset[1],offset[2])          #расчет желаемой координаты

    robot.MoveL(point)                                                  #линейное движение к определённой координате

def move\_c(speed, offset\_1, offset\_2):                                  #функция дугообразного движения

    robot.setSpeed(speed)                                               #задание скорости движения робота

    point1 = robot.Pose()\*transl(offset\_1[0],offset\_1[1],offset\_1[2])   #промежуточная координата

    point2 = robot.Pose()\*transl(offset\_2[0],offset\_2[1],offset\_2[2])   #конечная координата

    robot.MoveC(point1,point2)                                          #дугообразное движение к определённой координате

def move\_j(speed, offset):                                              #Функция ptp движения

    robot.setSpeed(speed)                                               #Задание скорости движения робота

    point = robot.Pose()\*transl(offset[0],offset[1],offset[2])          #расчет желаемой координаты

    robot.MoveJ(point)                                                  #ptp движение к определённой координате

def letter\_v(k):                                                        #функция написания буквы В

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (96\*k, 0, 0))                                           #линейное движение к В1

    move\_c(75,(-20\*k, -16\*k, 0) , (-40\*k, 0, 0))                        #circle движение В2 В3

    move\_c(75,(-28\*k, -32\*k, 0) , (-56\*k, 0, 0))                        #circle движение В4 В5

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

def letter\_a(k):                                                        #функция написания буквы А

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (96\*k, -16\*k, 0))                                       #линейные движения

    move\_l(75, (-96\*k, -16\*k, 0))

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

    move\_l(50, (36\*k, 6\*k, 0))

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (0, 20\*k, 0))

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

def letter\_d(k):                                                        #функция написания буквы Д

    move\_l(50, (0, 0, -50))

    move\_l(75, (12\*k, 0, 0))

    move\_l(75, (0, -32\*k, 0))

    move\_l(75, (-12\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (12\*k, 28\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, 0, 0))

    move\_l(75, (0, -24\*k, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_i(k):                                                        #функция написания буквы И

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(75, (96\*k, -32\*k, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_m(k):                                                        #функция написания буквы М

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, -8\*k, 0))

    move\_l(75, (-96\*k, -8\*k, 0))

    move\_l(75, (96\*k, -8\*k, 0))

    move\_l(75, (-96\*k, -8\*k, 0))

    move\_l(50, (0, 0, 50))

def letter\_o(k):                                                        #функция написания буквы О

    move\_l(50, (0, 0, -50))

    move\_c(75,(-16\*k, -16\*k, 0) , (0, -32\*k, 0))

    move\_l(75, (64\*k, 0, 0))

    move\_c(75,(16\*k, 16\*k, 0) , (0, 32\*k, 0))

    move\_l(75, (-64\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_t(k):                                                        #функция написания буквы Т

    move\_l(50, (0, 0, -50))

    move\_l(75, (0, -32\*k, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (0, 16, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_ya(k):                                                        #функция написания буквы Я

    move\_l(50, (0, 0, -50))

    move\_l(75, (48\*k, -32\*k, 0))

    move\_c(75,(24\*k, 24\*k, 0) , (48\*k, 0, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_n(k):                                                        #функция написания буквы Н

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (0, -32\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (48\*k, 32\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (0, -32\*k, 0))

    move\_l(50, (0, 0, 50))

def name\_surname\_turned():

    robot.setPoseFrame(RDK.Item('Frame2'))                              #выбор смещенной и повернутой системы координат

    scale=1

    robot.MoveJ(Workbench.Pose())                                       #движение в точку над Именем

    move\_j(100, (10, 200, -200))                                        #движение  точку над буквой

    letter\_v(scale)                                                     #отрисовка буквы В

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10, 140, -200))

    letter\_a(scale)                                                     #отрисовка буквы А

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (-5, 80, -200))

    letter\_d(scale)                                                     #отрисовка буквы Д

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10+96\*scale, 20, -200))

    letter\_i(scale)                                                     #отрисовка буквы И

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10, -40, -200))

    letter\_m(scale)                                                     #отрисовка буквы М

    robot.MoveJ(Workbench.Pose())

    robot.MoveJ(Workbench\_surname.Pose())                               #движение в точку над Фамилией

    move\_j(100, (60, 200, -200))                                        #движение  точку над буквой

    letter\_m(scale)                                                     #отрисовка буквы М

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60+16\*scale, 140, -200))

    letter\_o(scale)                                                     #отрисовка буквы О

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, 80, -200))

    letter\_v(scale)                                                     #отрисовка буквы В

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60+96\*scale, 20, -200))

    letter\_t(scale)                                                     #отрисовка буквы Т

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, -40, -200))

    letter\_ya(scale)                                                    #отрисовка буквы Я

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, -100, -200))

    letter\_n(scale)                                                     #отрисовка буквы Н

    robot.MoveJ(Workbench\_surname.Pose())

    robot.setPoseFrame(base\_frame)

    robot.MoveJ(Base.Pose())                                            #возврат в исходное положение

name\_surname\_turned()

**Scale**

from robolink import \*                                                  # RoboDK API

from robodk import \*                                                    # Robot toolbox

import tkinter as tk

from tkinter import \*

from tkinter import messagebox

#import numpy as np

RDK = Robolink()

robot = RDK.ItemUserPick('Select a robot', ITEM\_TYPE\_ROBOT)             #инициализация робота

if not robot.Valid():

    raise Exception('No robot selected or available')

robot.setPoseFrame(robot.PoseFrame())

robot.setPoseTool(robot.PoseTool())

robot.setRounding(10)

robot.setSpeed(100)

base\_frame = RDK.Item('KUKA KR 4 R600 Base')

Base = RDK.Item('Base')                                                 #определение домашнего положения

Workbench = RDK.Item('Workbench3')

Workbench\_surname = RDK.Item('Workbench\_surname3')

def move\_l(speed, offset):                                              #функция линейного движения

    robot.setSpeed(speed)                                               #задание скорости движения робота

    point = robot.Pose()\*transl(offset[0],offset[1],offset[2])          #расчет желаемой координаты

    robot.MoveL(point)                                                  #линейное движение к определённой координате

def move\_c(speed, offset\_1, offset\_2):                                  #функция дугообразного движения

    robot.setSpeed(speed)                                               #задание скорости движения робота

    point1 = robot.Pose()\*transl(offset\_1[0],offset\_1[1],offset\_1[2])   #промежуточная координата

    point2 = robot.Pose()\*transl(offset\_2[0],offset\_2[1],offset\_2[2])   #конечная координата

    robot.MoveC(point1,point2)                                          #дугообразное движение к определённой координате

def move\_j(speed, offset):                                              #Функция ptp движения

    robot.setSpeed(speed)                                               #Задание скорости движения робота

    point = robot.Pose()\*transl(offset[0],offset[1],offset[2])          #расчет желаемой координаты

    robot.MoveJ(point)                                                  #ptp движение к определённой координате

def letter\_v(k):                                                        #функция написания буквы В

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (96\*k, 0, 0))                                           #линейное движение к В1

    move\_c(75,(-20\*k, -16\*k, 0) , (-40\*k, 0, 0))                        #circle движение В2 В3

    move\_c(75,(-28\*k, -32\*k, 0) , (-56\*k, 0, 0))                        #circle движение В4 В5

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

def letter\_a(k):                                                        #функция написания буквы А

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (96\*k, -16\*k, 0))                                       #линейные движения

    move\_l(75, (-96\*k, -16\*k, 0))

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

    move\_l(50, (36\*k, 6\*k, 0))

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (0, 20\*k, 0))

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

def letter\_d(k):                                                        #функция написания буквы Д

    move\_l(50, (0, 0, -50))

    move\_l(75, (12\*k, 0, 0))

    move\_l(75, (0, -32\*k, 0))

    move\_l(75, (-12\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (12\*k, 28\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, 0, 0))

    move\_l(75, (0, -24\*k, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_i(k):                                                        #функция написания буквы И

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(75, (96\*k, -32\*k, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_m(k):                                                        #функция написания буквы М

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, -8\*k, 0))

    move\_l(75, (-96\*k, -8\*k, 0))

    move\_l(75, (96\*k, -8\*k, 0))

    move\_l(75, (-96\*k, -8\*k, 0))

    move\_l(50, (0, 0, 50))

def letter\_o(k):                                                        #функция написания буквы О

    move\_l(50, (0, 0, -50))

    move\_c(75,(-16\*k, -16\*k, 0) , (0, -32\*k, 0))

    move\_l(75, (64\*k, 0, 0))

    move\_c(75,(16\*k, 16\*k, 0) , (0, 32\*k, 0))

    move\_l(75, (-64\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_t(k):                                                        #функция написания буквы Т

    move\_l(50, (0, 0, -50))

    move\_l(75, (0, -32\*k, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (0, 16, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_ya(k):                                                        #функция написания буквы Я

    move\_l(50, (0, 0, -50))

    move\_l(75, (48\*k, -32\*k, 0))

    move\_c(75,(24\*k, 24\*k, 0) , (48\*k, 0, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_n(k):                                                        #функция написания буквы Н

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (0, -32\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (48\*k, 32\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (0, -32\*k, 0))

    move\_l(50, (0, 0, 50))

def task\_3(scale):

    robot.setPoseFrame(RDK.Item('Frame3'))

    robot.MoveJ(Workbench.Pose())                                       #движение в точку над Именем

    move\_j(100, (10, 200, -200))                                        #движение  точку над буквой

    letter\_v(scale)                                                     #отрисовка буквы В

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10, 140, -200))

    letter\_a(scale)                                                     #отрисовка буквы А

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (-5, 80, -200))

    letter\_d(scale)                                                     #отрисовка буквы Д

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10+96\*scale, 20, -200))

    letter\_i(scale)                                                     #отрисовка буквы И

    robot.MoveJ(Workbench.Pose())

    move\_j(100, (10, -40, -200))

    letter\_m(scale)                                                     #отрисовка буквы М

    robot.MoveJ(Workbench.Pose())

    robot.MoveJ(Workbench\_surname.Pose())                               #движение в точку над Фамилией

    move\_j(100, (60, 200, -200))                                        #движение  точку над буквой

    letter\_m(scale)                                                     #отрисовка буквы М

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60+16\*scale, 140, -200))

    letter\_o(scale)                                                     #отрисовка буквы О

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, 80, -200))

    letter\_v(scale)                                                     #отрисовка буквы В

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60+96\*scale, 20, -200))

    letter\_t(scale)                                                     #отрисовка буквы Т

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, -40, -200))

    letter\_ya(scale)                                                    #отрисовка буквы Я

    robot.MoveJ(Workbench\_surname.Pose())

    move\_j(100, (60, -100, -200))

    letter\_n(scale)                                                     #отрисовка буквы Н

    robot.MoveJ(Workbench\_surname.Pose())

    robot.setPoseFrame(base\_frame)

    robot.MoveJ(Base.Pose())                                            #возврат в исходное положение

scale = 0.0

def choose\_scale():                                                     #функция вывода окна для обработки вносимых значений

    """Функция вывода окна для обработки вносимых значений."""

    window = Tk()

    window.title("Напишите варианты")

    window.geometry("400x100")

    frame = Frame(window, padx=10, pady=10)

    frame.pack(expand=True)

    scale\_lbl = Label(frame, text="Укажите масштаб")

    scale\_lbl.grid(row=1, column=1, pady=10)

    scale\_ent = Entry(frame)

    scale\_ent.grid(row=1, column=2)

    scale\_ent.insert(0, "1")

    def handle\_add\_button():

global scale

      try:

          scale = float(scale\_ent.get())

          window.destroy() # Закрываем окно

      except ValueError:

           messagebox.showerror("Ошибка", "Введите корректное число.")

    scale\_btn = Button(

        frame,

        text="Добавить",

        command=handle\_add\_button  # Измененная команда

    )

    scale\_btn.grid(row=1, column=3)

    window.mainloop()

choose\_scale()

task\_3(scale)  # Запуск task\_3

**Order**

from robolink import \*                                                  # RoboDK API

from robodk import \*                                                    # Robot toolbox

import tkinter as tk

from tkinter import \*

from tkinter import messagebox

#import numpy as np

RDK = Robolink()

robot = RDK.ItemUserPick('Select a robot', ITEM\_TYPE\_ROBOT)             #инициализация робота

if not robot.Valid():

    raise Exception('No robot selected or available')

robot.setPoseFrame(robot.PoseFrame())

robot.setPoseTool(robot.PoseTool())

robot.setRounding(10)

robot.setSpeed(100)

base\_frame = RDK.Item('KUKA KR 4 R600 Base')

Base = RDK.Item('Base')                                                 #определение домашнего положения

Workbench = RDK.Item('Workbench4')

Workbench\_surname = RDK.Item('Workbench\_surname4')

def move\_l(speed, offset):                                              #функция линейного движения

    robot.setSpeed(speed)                                               #задание скорости движения робота

    point = robot.Pose()\*transl(offset[0],offset[1],offset[2])          #расчет желаемой координаты

    robot.MoveL(point)                                                  #линейное движение к определённой координате

def move\_c(speed, offset\_1, offset\_2):                                  #функция дугообразного движения

    robot.setSpeed(speed)                                               #задание скорости движения робота

    point1 = robot.Pose()\*transl(offset\_1[0],offset\_1[1],offset\_1[2])   #промежуточная координата

    point2 = robot.Pose()\*transl(offset\_2[0],offset\_2[1],offset\_2[2])   #конечная координата

    robot.MoveC(point1,point2)                                          #дугообразное движение к определённой координате

def move\_j(speed, offset):                                              #Функция ptp движения

    robot.setSpeed(speed)                                               #Задание скорости движения робота

    point = robot.Pose()\*transl(offset[0],offset[1],offset[2])          #расчет желаемой координаты

    robot.MoveJ(point)                                                  #ptp движение к определённой координате

def letter\_v(k):                                                        #функция написания буквы В

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (96\*k, 0, 0))                                           #линейное движение к В1

    move\_c(75,(-20\*k, -16\*k, 0) , (-40\*k, 0, 0))                        #circle движение В2 В3

    move\_c(75,(-28\*k, -32\*k, 0) , (-56\*k, 0, 0))                        #circle движение В4 В5

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

def letter\_a(k):                                                        #функция написания буквы А

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (96\*k, -16\*k, 0))                                       #линейные движения

    move\_l(75, (-96\*k, -16\*k, 0))

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

    move\_l(50, (36\*k, 6\*k, 0))

    move\_l(50, (0, 0, -50))                                             #опускание маркера на рабочую поверхность

    move\_l(75, (0, 20\*k, 0))

    move\_l(50, (0, 0, 50))                                              #подьём маркера над рабочей поверхностью

def letter\_d(k):                                                        #функция написания буквы Д

    move\_l(50, (0, 0, -50))

    move\_l(75, (12\*k, 0, 0))

    move\_l(75, (0, -32\*k, 0))

    move\_l(75, (-12\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (12\*k, 28\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, 0, 0))

    move\_l(75, (0, -24\*k, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_i(k):                                                        #функция написания буквы И

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(75, (96\*k, -32\*k, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_m(k):                                                        #функция написания буквы М

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, -8\*k, 0))

    move\_l(75, (-96\*k, -8\*k, 0))

    move\_l(75, (96\*k, -8\*k, 0))

    move\_l(75, (-96\*k, -8\*k, 0))

    move\_l(50, (0, 0, 50))

def letter\_o(k):                                                        #функция написания буквы О

    move\_l(50, (0, 0, -50))

    move\_c(75,(-16\*k, -16\*k, 0) , (0, -32\*k, 0))

    move\_l(75, (64\*k, 0, 0))

    move\_c(75,(16\*k, 16\*k, 0) , (0, 32\*k, 0))

    move\_l(75, (-64\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_t(k):                                                        #функция написания буквы Т

    move\_l(50, (0, 0, -50))

    move\_l(75, (0, -32\*k, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (0, 16, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_ya(k):                                                        #функция написания буквы Я

    move\_l(50, (0, 0, -50))

    move\_l(75, (48\*k, -32\*k, 0))

    move\_c(75,(24\*k, 24\*k, 0) , (48\*k, 0, 0))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

def letter\_n(k):                                                        #функция написания буквы Н

    move\_l(50, (0, 0, -50))

    move\_l(75, (96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (0, -32\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (-96\*k, 0, 0))

    move\_l(50, (0, 0, 50))

    move\_l(50, (48\*k, 32\*k, 0))

    move\_l(50, (0, 0, -50))

    move\_l(75, (0, -32\*k, 0))

    move\_l(50, (0, 0, 50))

def draw\_letter\_name(position, i, scale):                               #функция перестановки букв в имени (пользователь сам выбирает последовательность)

    if i == 1:                                                          #в зависимости от позиции задаем движение к точке над буквой (этот цикл для первой позиции в имени)

        if position[i] in [1, 2, 5]:                                    #в зависимости от буквы задаем движение к точке над буквой

            move\_j(100, (10, 200, -200))                                #если нужно нарисовать В,А,М

        elif position[i] == 3:                                          #если нужно нарисовать Д

            move\_j(100, (-5, 200, -200))

        elif position[i] == 4:                                          #если нужно нарисовать И

            move\_j(100, (10+96\*scale, 200, -200))

    if i == 2:                                                          #для второй позиции

        if position[i] in [1, 2, 5]:

            move\_j(100, (10, 140, -200))

        elif position[i] == 3:

            move\_j(100, (-5, 140, -200))

        elif position[i] == 4:

            move\_j(100, (10+96\*scale, 140, -200))

    if i == 3:                                                          #для третей позиции

        if position[i] in [1, 2, 5]:

            move\_j(100, (10, 80, -200))

        elif position[i] == 3:

            move\_j(100, (-5, 80, -200))

        elif position[i] == 4:

            move\_j(100, (10+96\*scale, 80, -200))

    if i == 4:                                                          #для четвертой позиции

        if position[i] in [1, 2, 5]:

            move\_j(100, (10, 20, -200))

        elif position[i] == 3:

            move\_j(100, (-5, 20, -200))

        elif position[i] == 4:

            move\_j(100, (10+96\*scale, 20, -200))

    if i == 5:                                                          #для пятой позиции

        if position[i] in [1, 2, 5]:

            move\_j(100, (10, -40, -200))

        elif position[i] == 3:

            move\_j(100, (-5, -40, -200))

        elif position[i] == 4:

            move\_j(100, (10+96\*scale, -40, -200))

    if position[i] == 1:                                                #цифре 1 соответствует буква В

        letter\_v(scale)

    if position[i] == 2:                                                #цифре 2 соответствует буква А

        letter\_a(scale)

    if position[i] == 3:                                                #цифре 3 соответствует буква Д

        letter\_d(scale)

    if position[i] == 4:                                                #цифре 4 соответствует буква И

        letter\_i(scale)

    if position[i] == 5:                                                #цифре 5 соответствует буква М

        letter\_m(scale)

def draw\_letter\_surname(position, i, scale):                            #функция перестановки букв в фамилии (пользователь сам выбирает последовательность)

    if i == 1:                                                          #в зависимости от позиции задаем движение к точке над буквой (этот цикл для первой позиции в фамилии)

        if position[i] in [1, 3, 5, 6]:                                 #в зависимости от буквы задаем движение к точке над буквой

            move\_j(100, (60, 200, -200))                                #если нужно нарисовать М, В, Я, Н

        elif position[i] == 2:                                          #если нужно нарисовать О

            move\_j(100, (60+16\*scale, 200, -200))

        elif position[i] == 4:                                          #если нужно нарисовать Т

            move\_j(100, (60+96\*scale, 200, -200))

    if i == 2:                                                          #для второй позиции

        if position[i] in [1, 3, 5, 6]:

            move\_j(100, (60, 140, -200))

        elif position[i] == 2:

            move\_j(100, (60+16\*scale, 140, -200))

        elif position[i] == 4:

            move\_j(100, (60+96\*scale, 140, -200))

    if i == 3:                                                          #для третей позиции

        if position[i] in [1, 3, 5, 6]:

            move\_j(100, (60, 80, -200))

        elif position[i] == 2:

            move\_j(100, (60+16\*scale, 80, -200))

        elif position[i] == 4:

            move\_j(100, (60+96\*scale, 80, -200))

    if i == 4:                                                          #для четвертой позиции

        if position[i] in [1, 3, 5, 6]:

            move\_j(100, (60, 20, -200))

        elif position[i] == 2:

            move\_j(100, (60+16\*scale, 20, -200))

        elif position[i] == 4:

            move\_j(100, (60+96\*scale, 20, -200))

    if i == 5:                                                          #для пятой позиции

        if position[i] in [1, 3, 5, 6]:

            move\_j(100, (60, -40, -200))

        elif position[i] == 2:

            move\_j(100, (60+16\*scale, -40, -200))

        elif position[i] == 4:

            move\_j(100, (60+96\*scale, -40, -200))

    if i == 6:                                                          #для шестой позиции

        if position[i] in [1, 3, 5, 6]:

            move\_j(100, (60, -100, -200))

        elif position[i] == 2:

            move\_j(100, (60+16\*scale, -100, -200))

        elif position[i] == 4:

            move\_j(100, (60+96\*scale, -100, -200))

    if position[i] == 1:                                                #цифре 1 соответствует буква М

        letter\_m(scale)

    if position[i] == 2:                                                #цифре 2 соответствует буква О

        letter\_o(scale)

    if position[i] == 3:                                                #цифре 3 соответствует буква В

        letter\_v(scale)

    if position[i] == 4:                                                #цифре 4 соответствует буква Т

        letter\_t(scale)

    if position[i] == 5:                                                #цифре 5 соответствует буква Я

        letter\_ya(scale)

    if position[i] == 6:                                                #цифре 6 соответствует буква Н

        letter\_n(scale)

def task\_4(name, surname):

    robot.setPoseFrame(RDK.Item('Frame4'))

    k=1

    order\_name=5

    position\_name=[0,0,0,0,0,0]

    Number\_name=float(name)

    for i in range(order\_name):                                         #алгоритм перевода 5ти значного числа (в имени) в массив из входящих в него цифр

        position\_name[i+1]=(Number\_name//10\*\*(order\_name-(i+1)))

        Number\_name%=10 \*\* (order\_name-(i+1))

    robot.MoveJ(Workbench.Pose())                                       #движение в точку над именем

    draw\_letter\_name(position\_name, 1, k)                               #отрисовка буквы в имени на первой позиции

    robot.MoveJ(Workbench.Pose())

    draw\_letter\_name(position\_name, 2, k)                               #отрисовка буквы на второй позиции

    robot.MoveJ(Workbench.Pose())

    draw\_letter\_name(position\_name, 3, k)                               #отрисовка буквы на третей позиции

    robot.MoveJ(Workbench.Pose())

    draw\_letter\_name(position\_name, 4, k)                               #отрисовка буквы на четвертой позиции

    robot.MoveJ(Workbench.Pose())

    draw\_letter\_name(position\_name, 5, k)                               #отрисовка буквы на пятой позиции

    robot.MoveJ(Workbench.Pose())

    order\_surname=6

    position\_surname=[0,0,0,0,0,0,0]

    Number\_surname=float(surname)

    for i in range(order\_surname):                                      #алгоритм перевода 6ти значного числа (в фамилии) в массив из входящих в него цифр

        position\_surname[i+1]=(Number\_surname//10\*\*(order\_surname-(i+1)))

        Number\_surname%=10 \*\* (order\_surname-(i+1))

    robot.MoveJ(Workbench\_surname.Pose())                               #движение в точку над фамилией

    draw\_letter\_surname(position\_surname, 1, k)                         #отрисовка буквы в фамилии на первой позиции

    robot.MoveJ(Workbench\_surname.Pose())

    draw\_letter\_surname(position\_surname, 2, k)                         #отрисовка буквы на второй позиции

    robot.MoveJ(Workbench\_surname.Pose())

    draw\_letter\_surname(position\_surname, 3, k)                         #отрисовка буквы на третей позиции

    robot.MoveJ(Workbench\_surname.Pose())

    draw\_letter\_surname(position\_surname, 4, k)                         #отрисовка буквы на четвертой позиции

    robot.MoveJ(Workbench\_surname.Pose())

    draw\_letter\_surname(position\_surname, 5, k)                         #отрисовка буквы на пятой позиции

    robot.MoveJ(Workbench\_surname.Pose())

    draw\_letter\_surname(position\_surname, 6, k)                         #отрисовка буквы на шестой позиции

    robot.MoveJ(Workbench\_surname.Pose())

    robot.setPoseFrame(base\_frame)

    robot.MoveJ(Base.Pose())

def choose\_scale():

    """Функция вывода окна для обработки вносимых значений."""

    window = Tk()

    window.title("Напишите варианты")

    window.geometry("450x150")

    frame = Frame(window, padx=10, pady=10)

    frame.pack(expand=True)

    name\_order\_lbl = Label(frame, text="Укажите порядок букв в имени")

    name\_order\_lbl.grid(row=2, column=1, pady=10)

    name\_order\_ent = Entry(frame)

    name\_order\_ent.grid(row=2, column=2)

    name\_order\_ent.insert(0, "12345")

    surname\_order\_lbl = Label(frame, text="Укажите порядок букв в фамилии")

    surname\_order\_lbl.grid(row=3, column=1, pady=10)

    surname\_order\_ent = Entry(frame)

    surname\_order\_ent.grid(row=3, column=2)

    surname\_order\_ent.insert(0, "123456")

    def handle\_draw\_button():

        try:

            name\_order = name\_order\_ent.get()

            surname\_order = surname\_order\_ent.get()

            window.destroy() # Закрываем окно

            task\_4(name\_order, surname\_order)  # Запускаем task\_4

        except Exception as e:

           messagebox.showerror("Ошибка", f"Ошибка: {e}")

    name\_order\_btn = Button(

        frame,

        text="Нарисовать",

        command=handle\_draw\_button  # Измененная команда

    )

    name\_order\_btn.grid(row=3, column=3)

    window.mainloop()

choose\_scale()