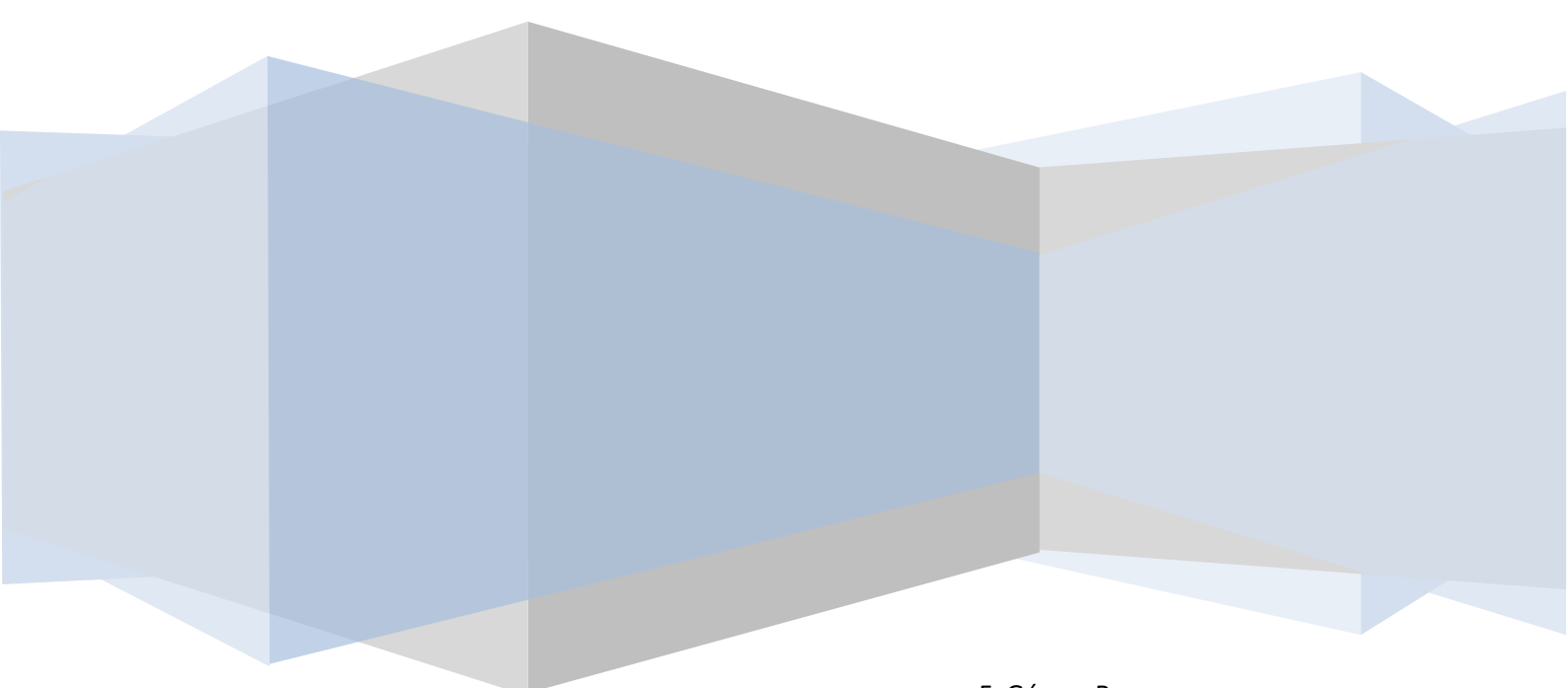


ROBÓTICA
CUARTO CURSO DEL GRADO EN
INGENIERÍA INFORMÁTICA



Práctica 2.2

Programación y Ejecución de tareas en un
Manipulador Robótico Colaborativo.



ENUNCIADO

Introducción

Esta práctica es el colofón de lo aprendido en las prácticas anteriores. Vamos a llevar a la realidad los programas testados en RoboDK. Una vez aprendido el manejo de la interfaz de RoboDK y su utilización para simular la ejecución del programa en nuestro robot, se procederá a programar distintas tareas basadas en la aplicación del procedimiento de 'pick & place' utilizado con anterioridad.

Recordad que para ejecutar vuestros programas en un robot concreto tenéis que ejecutar el comando 'Iniciación' de la siguiente forma. Para el robot 1:

```
Identificador = Iniciacion('Robot_1',codigo) .
```

Para el robot 2:

```
Identificador = Iniciacion('Robot_2',codigo)
```

Al principio de la práctica se asignará a cada grupo un robot con el que trabajar y un código en particular. Cuando por parte del profesor se os indique que vuestro código está activo en el robot, será posible la ejecución de vuestros programas en el mismo. Mientras vuestro código no esté activo la conexión con el robot no será posible.

Cada grupo tendrá preparados, funcionando, todos los apartados correspondientes a la práctica anterior. El profesor sugerirá una modificación de alguno de los resultados presentados. La modificación se preparará haciéndola funcionar en RoboDK, y se presentará al profesor. Cuando éste haya habilitado en el robot el código asignado, el programa se ejecutará en el robot. Para considerar apta esta práctica, ha de funcionar en el robot al menos dos de los apartados programados en la práctica anterior.

Ejercicios

En primer lugar, se propone la programación de una tarea de **apilado y desapilado**, para continuar con una tarea de **empaquetado**.. Para la realización de los ejercicios de empaquetado y ensamblado se introduce una nueva función cuyo prototipo es:

```
MoveL_Robot_lab(angulos, aceleracion, velocidad, Identificador, codigo)
```

El uso de esta función es similar al descrito en la práctica 4 para la función "MoveJ_Robot_lab". La diferencia estriba en el comportamiento del robot. En este caso el

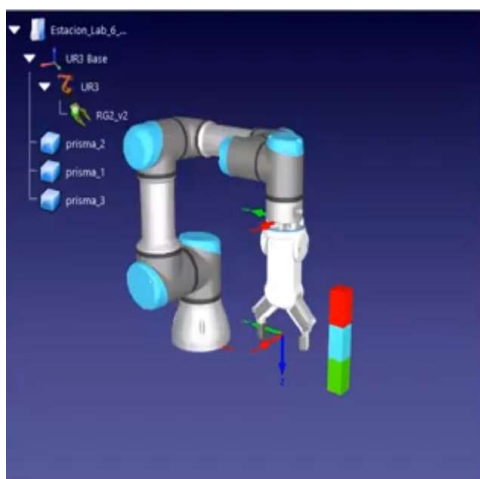
movimiento de las articulaciones del robot se realiza de forma que el TCP describe una línea recta para moverse entre la configuración de origen y la de destino.

Todos los ejercicios se realizarán primero en simulación y posteriormente se testarán en el robot real.

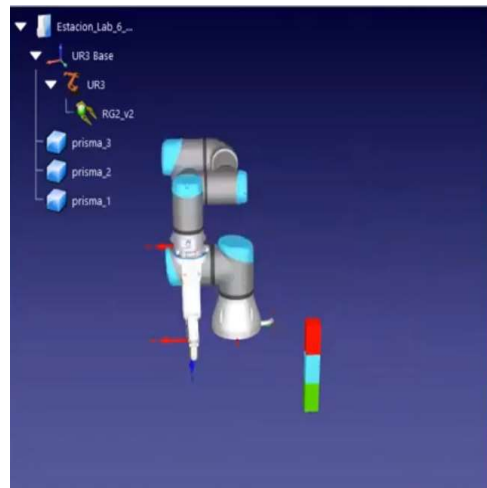
Ejercicio 1

Modifique el escenario de robot del fichero 'Estacion_Lab_6_2.rdk' según se observa en la figura 1 a) y sávelo con otro nombre. Programe una rutina para que el manipulador desapile y apile las piezas según se observa en la figura 1b). Para tener una idea de cómo es la operación que hay que realizar visualice el vídeo:

https://www.youtube.com/watch?v=76_wGcNYY84



a)



b)

Figura 1

La nueva torre ha de apilarse sobre el punto (35, 0, 0)

El código que resulte de la programación de esta rutina deberá mantener la misma estructura del programado en la práctica 4, modificando solo los aspectos necesarios de la actualización de configuraciones y añadiendo el cálculo de la configuración auxiliar que se utiliza para reposar las piezas intermedias, antes de volver a realizar el apilado. Observe que el orden de apilado de la columna final es el mismo que el de la columna inicial.

Como sugerencia para alcanzar la solución definitiva se propone realizar primero un apilado donde el orden de los bloques en la columna final sea inverso al orden en la columna

inicial. A partir del código generado para realizar esa rutina, es fácil desarrollar el programa final solicitado.

Ejercicio 2

Descargue el fichero 'Estacion_Lab_6_2_caja.rdk' y ábralo en RoboDK, deberá aparecer un escenario como el que se muestra en la figura 2 a). Se pretende programar una rutina de empaquetado de las seis piezas que han de quedar agrupadas dentro de la caja tal y como se observa en la figura 2 b). Observe la ubicación del sistema de referencia de la caja. Su configuración es: posición (20, 5, 0) y ángulos de Euler ZYZ (0, 0, $-\pi/4$). La configuración de la primera pieza que se cogerá es: posición (20, -10, 0) y ángulos de Euler ZYZ (0, 0, 0), el resto de las piezas de la primera fila están separadas por 10 cm a lo largo del eje 'X'. La segunda fila de piezas está separada de la primera por 5 cm medidos en la dirección del eje 'Y'.

Para tener una idea de cómo es la operación a realizar visualice el vídeo:

<https://youtu.be/VsMKOI7iGQU>

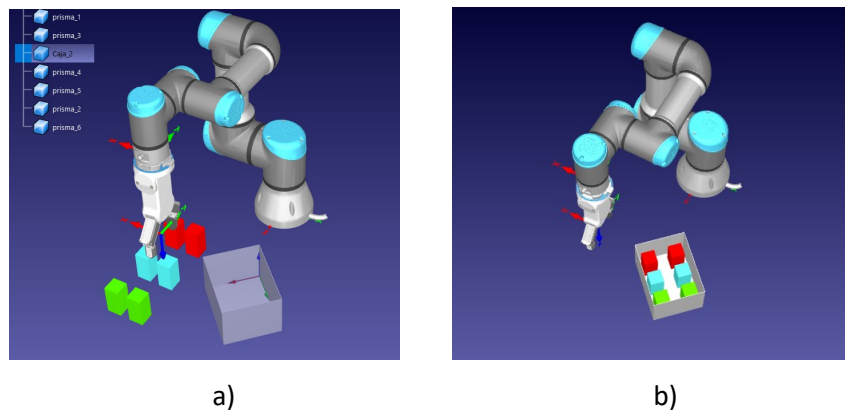


Figura 2

La disposición en la que han de quedar los bloques dentro de la caja se muestra en la figura 3.

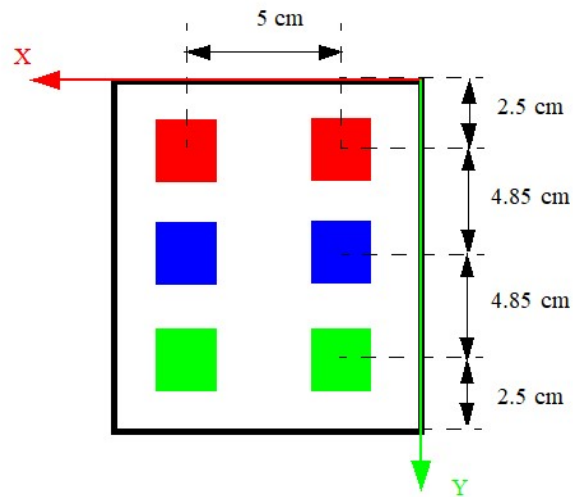


Figura 3

Observe en el video que, ahora, en las maniobras de aproximación y despegue el TCP realiza trayectorias rectas, por lo que para ejecutar estos movimientos hay que utilizar el comando "MoveL_Robot_lab". Esto es especialmente necesario en la maniobra de place, para evitar colisionar con la caja en el momento de manipular dentro de ella.