



Universitat d'Alacant  
Universidad de Alicante

Grado en Ingeniería Robótica  
Sistemas Multirobot

## **Práctica ABB**

Autor:

Miguel Ferrer Castellá

# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Comportamiento 1</b>	<b>2</b>
2.1. Objetivo . . . . .	2
2.2. Escena . . . . .	2
2.3. Simulación . . . . .	3
<b>3. Comportamiento 2</b>	<b>3</b>
3.1. Objetivo . . . . .	3
3.2. Escena . . . . .	3
3.3. Lógica de estación . . . . .	4
3.4. Programación en <i>RAPID</i> . . . . .	4
3.5. Simulación . . . . .	5
<b>4. Comportamiento 3</b>	<b>5</b>
4.1. Objetivo . . . . .	5
4.2. Escena . . . . .	5
4.3. Lógica de estación . . . . .	6
4.4. Programación en <i>RAPID</i> . . . . .	7
4.5. Simulación . . . . .	7
<b>5. Conclusiones</b>	<b>7</b>

# 1. Introducción

El presente informe expone el desarrollo que aborda la programación y simulación de robots ABB IRB120 para la ejecución de distintas tareas colaborativas y de manipulación. En concreto, se han configurado tres comportamientos o rutinas distintas, que van desde un simple recorrido sobre una mesa de trabajo, hasta la sincronización de dos robots que operan de manera coordinada para sostener y desplazar un objeto compartido, y finalizan con la compleja labor de apilar doce piezas con la ayuda de ventosas inteligentes. Dichas acciones se han diseñado a través del lenguaje de programación *RAPID* y se han puesto a prueba en el entorno de RobotStudio, permitiendo comprobar la validez tanto de la lógica de estación como de los movimientos planificados.

En este contexto, cada comportamiento plantea un reto incremental de integración de hardware y software: la introducción de manipuladores de vacío controlados por señales digitales para sincronizar las operaciones entre robots, y la verificación de la orientación relativa de las herramientas en el espacio.

## 2. Comportamiento 1

### 2.1. Objetivo

El primer comportamiento se trata de realizar una trayectoria simple sobre una mesa de trabajo, como se muestra en el vídeo siguiente.

### 2.2. Escena

En la escena se posicionan en  $(0,0,0)$  y  $(0,800,0)$  dos robots ABB IRB 120. Además, se ha modulado una mesa de trabajo sobre la que se realizarán los movimientos.

Cada robot se controla con una controladora diferente, con la que se crea una trayectoria con movimientos lineales, empezando en una posición de partida, realizando un cuadrado sobre la superficie de la mesa de trabajo, y terminando en la posición de partida.

Ambos robots realizan las trayectorias simultáneamente, aunque no están comunicándose mediante la Lógica de estación.

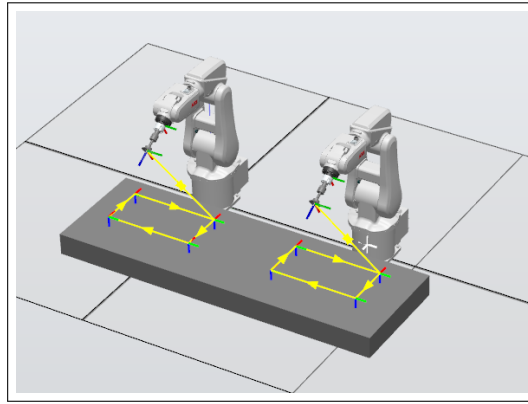


Figura 1: Escena del comportamiento 1.

### 2.3. Simulación

Se muestra en el siguiente enlace a Youtube la simulación de las trayectorias llevadas a cabo.

## 3. Comportamiento 2

### 3.1. Objetivo

En este segundo comportamiento se hará uso de una ventosa inteligente obtenida de internet utilizada como manipuladores finales de los robots ABB IRB 120. Los robots sostendrán entre los dos una caja posicionada entre ellos, tal y como se muestra en un vídeo de ejemplo visible en este enlace.

### 3.2. Escena

La escena consta de los dos robots ABB IRB 120 posicionados como en el comportamiento 1. Una caja se encuentra posicionada en el medio de los dos robots y sobre una mesa de trabajo anclada al mundo.

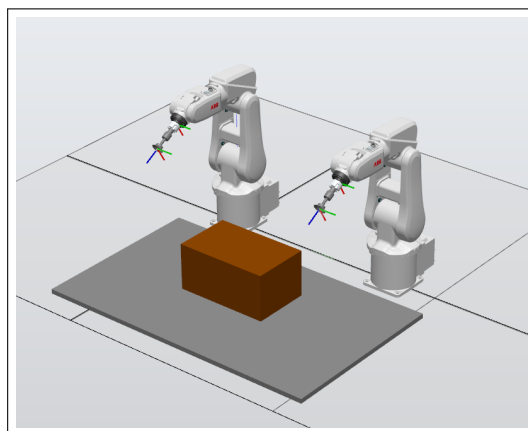


Figura 2: Escena del comportamiento 2.

Sólo hay 4 puntos definidos en la escena, los de inicio y fin de posición y los dos restantes, son uno en cada cara lateral de la caja, para que ambos robots puedan coger la caja por dichas caras.

### 3.3. Lógica de estación

En esta estación, las señales de los dos robots se combinan mediante una compuerta lógica AND. Cuando ambas condiciones **ready** están activas a la vez, el resultado de la compuerta habilita la señal de vacío de las ventosas con la señal **attach**. De esta manera, solo si las dos señales requeridas de los robots están presentes, se activa la alimentación de vacío.

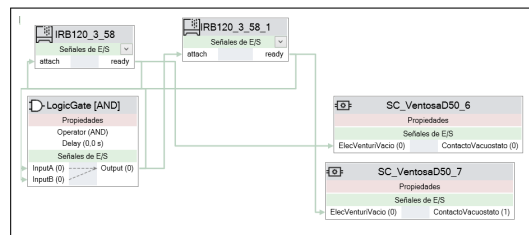


Figura 3: Lógica de estación del comportamiento 2.

### 3.4. Programación en *RAPID*

Este programa en *RAPID* consta de dos módulos, uno para cada robot **IRB120**, en los que se definen un punto de inicio (**Inicio**) y una posición de referencia (**Target\_10**). Cada robot comienza su rutina situándose en **Inicio**, comprueba el estado de las señales digitales (**ready** y **attach**) y, tras asegurarse de que puede actuar, se desplaza con un movimiento articular hasta la posición objetivo.

Una vez allí, el robot vuelve a modificar la señal **ready** y espera a que **attach** se active para iniciar la secuencia de movimientos lineales (**MoveL**) que desplazan la caja por el espacio. Dichos desplazamientos se llevan a cabo con **RelTool**, tomando el eje de la herramienta como referencia. Además, se incluyen pequeñas esperas de un segundo a través de **WaitTime**, dando tiempo al robot para estabilizar su posición y a la lógica de control para sincronizar otros eventos pertinentes.

Finalmente, cada robot desactiva de nuevo la señal **ready**, verifica **attach** y regresa a la posición de **Inicio**.

La diferencia clave entre ambos módulos radica en la orientación opuesta de las ventosas de cada robot, lo cual implica un eje X local invertido. Por ello, si se desea que ambos se muevan en la misma dirección en el sistema de coordenadas global, cada uno debe ejecutar un movimiento en sentido contrario respecto a su propio eje X local. Esta inversión de ejes se evidencia en los valores positivos y negativos de **RelTool**, permitiendo así que, a pesar de actuar de manera opuesta a nivel local, ambos robots realicen un movimiento convergente y coordinado para manipular la caja de forma conjunta.

### 3.5. Simulación

Se muestra la ejecución de la simulación en el siguiente enlace.

## 4. Comportamiento 3

### 4.1. Objetivo

El objetivo de este ultimo comportamiento propuesto, es el de apilar 12 piezas prismáticas, formando 4 pisos compuesto por 3 de estas piezas cada uno, cambiando de orientación en  $90^\circ$  en cada piso. Se trata de un clásico juego en el que se contruye esta torre, y en cada turno de jugador se retira una pieza de la torre y pierde el que derrumba la estructura.

Se puede observar el comportamiento que se pretende realizar en el siguiente enlace.

### 4.2. Escena

Como en los anteriores comportamientos, se posicionan dos robots ABB IRB 120. Las 12 piezas aparecen en 2 filas sobre una mesa de trabajo, las piezas se dividen en 2 materiales, para diferenciar las que coloca uno y otro robot. Estos objetos son dinámicos, por lo que si hubiera una colisión entre ellos o con los propios robots o mesa de trabajo influenciaría en el resultado esperado.

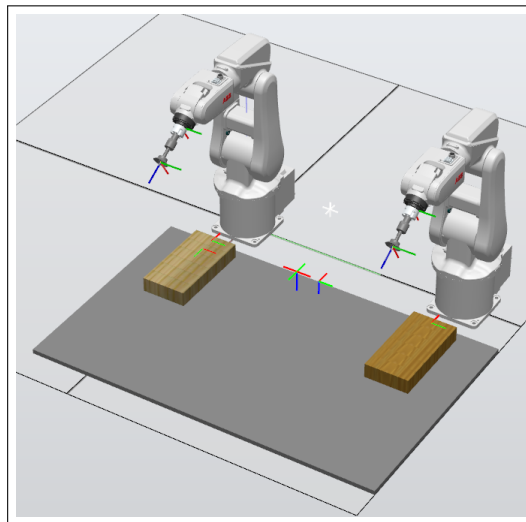


Figura 4: Escena del comportamiento 3.

Los puntos definidos para cada robot constan de la posición de partida, uno sobre la primera pieza de las 6 de cada fila, un segundo en la segunda fila y un último en la 3a fila.

La estructura final que se formará es la siguiente:

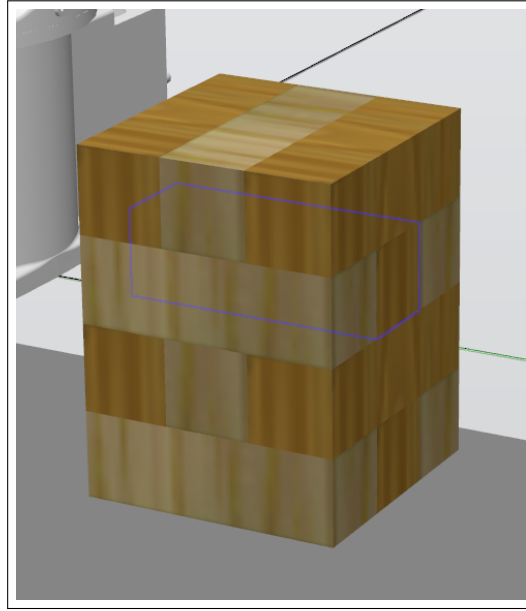


Figura 5: Estructura final.

### 4.3. Lógica de estación

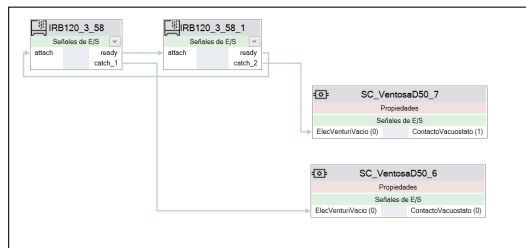


Figura 6: Lógica de estación del comportamiento 3.

La lógica de la estación se fundamenta en la interacción de los dos robots con sus respectivas ventosas, de modo que cada robot, antes de activar el vacío y asir el objeto, revisa si cumple las condiciones adecuadas de seguridad y sincronización con el otro robot.

De esta manera, cada robot dispone de ciertas señales de salida y de entrada que se relacionan con su estado de **ready**, el indicador de **attach** que determina si el objeto está sujeto y las órdenes de **catch\_1** o **catch\_2** que activan las ventosas. A su vez, las ventosas cuentan con una señal que enciende el dispositivo de vacío (**ElecVenturiVacío**), así como un **ContactoVacuostato** que confirma la presión de vacío correcta.

La secuencia se desarrolla del siguiente modo: cada robot anuncia que se encuentra listo mediante **ready**, verifica que el otro robot no esté generando conflicto con la señal **attach** y entonces activa la señal de vacío, iniciando la succión en la ventosa. Mientras el robot sujeta el objeto, se mantiene el vacío activo hasta que concluye la maniobra requerida y se desactiva **attach**, momento en que el robot detiene la generación de vacío y anuncia de nuevo su estado de **ready**. Con esto, la estación

garantiza que sólo se apliquen vacío y movimientos de agarre cuando cada robot haya comprobado que es seguro y no interfiera con la operación del otro robot, evitando así situaciones de error o caída del objeto.

#### 4.4. Programación en *RAPID*

En el módulo del robot IRB120\_3\_58, se definen dos variables (PX y PZ) que sirven para desplazar de manera incremental la ruta de agarre y depósito de los objetos. El robot comienza moviéndose a la posición *Inicio* y, dentro de un bucle, ejecuta sucesivas acciones de agarre y liberación. En cada ciclo, se mueve hacia *Target\_10*, para coger una pieza activando *catch\_1*, y luego se desplaza hacia *Target\_20* con las coordenadas ajustadas por PX y PZ apagando *catch\_1* para soltar la pieza, finalmente, regresa a una posición de seguridad. Cada vez que el robot finaliza un bloque de agarre y depósito, habilita la señal *ready* y aguarda *attach* para confirmar que el otro robot también ha completado su acción correspondiente. Este flujo se repite, incrementando PX, para ir cogiendo las piezas y PZ para ubicarlas en el siguiente piso. Cuando el bucle concluye, el robot retorna a *Inicio*.

Por otro lado, el módulo del robot IRB120\_3\_58\_1 describe una lógica muy similar, con la diferencia de que empieza a la espera de la señal *attach*, esperando a intervenir en segunda instancia, después de IRB120\_3\_58. El robot sigue la misma lógica, pero en vez de activar y desactivar *catch\_1*, lo hace con *catch\_2* para sujetar y soltar el objeto.

Se utiliza para todas las posiciones de coger o soltar, una posición previa de 100 o 125 mm superiores en el eje Z, para realizar un descenso lineal, asegurando un agarre exitoso.

Ambos robots coordinan sus acciones, de manera que cuando uno está en proceso de agarre o liberación, el otro aguarda la señal oportuna *ready* y la confirmación de *attach* para no interferir en la rutina.

#### 4.5. Simulación

El comportamiento completo se puede observar en este enlace, dónde se forma la estructura de 12 piezas de forma exitosa sin observar ninguna colisión entre las piezas.

### 5. Conclusiones

La práctica ha demostrado la eficacia de la planificación colaborativa en *RAPID* y la utilidad de las herramientas de simulación para anticipar posibles colisiones o imprecisiones de posicionamiento. La correcta implementación de la lógica de estación, basada en la integración de señales digitales y la verificación de estados, ha sido fundamental para asegurar un desarrollo fluido de cada maniobra.



Además, el tercer comportamiento, enfocado a la construcción de una torre con doce piezas, pone de manifiesto la importancia de la planificación de trayectorias y la correcta transición de alturas a fin de evitar choques entre los elementos y garantizar la repetibilidad de los movimientos. En conjunto, el trabajo realizado refleja la relevancia de la sincronización multirobot y el control distribuido, técnicas que resultan esenciales en aplicaciones industriales donde múltiples manipuladores deben cooperar de forma segura y eficiente. Se concluye, por tanto, que el dominio de la programación en *RAPID* y la comprensión de los principios de integración constituyen competencias claves para el diseño de entornos productivos flexibles y colaborativos.