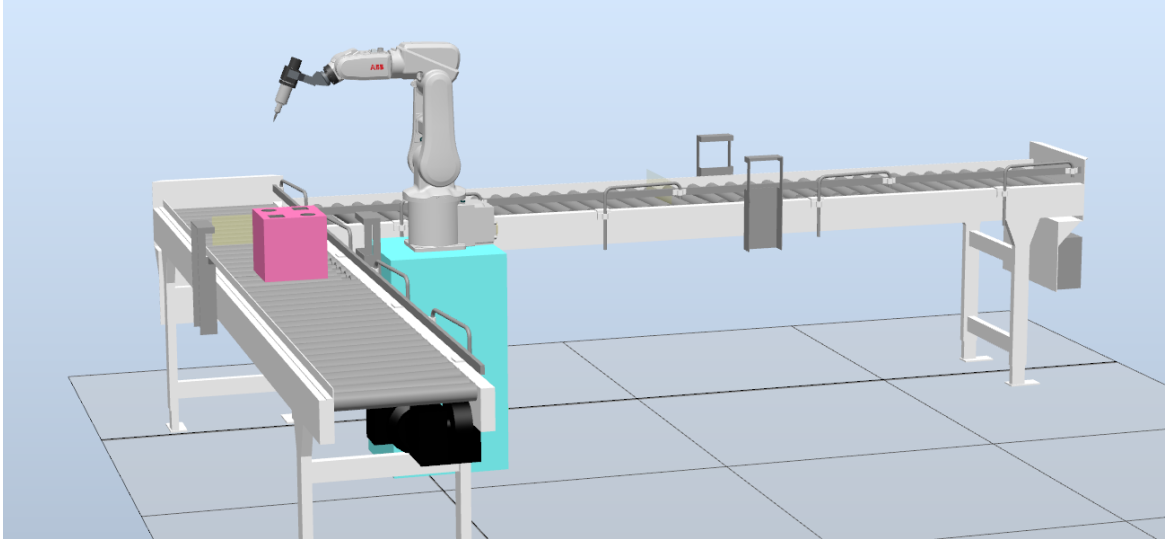


PARTE 1: UN CONTROLADOR Y UN ROBOT

Para realizar este trabajo he escogido el **robot IRB 120** de la biblioteca del ABB, este robot realizará los siguientes movimientos con la herramienta **myTool**:

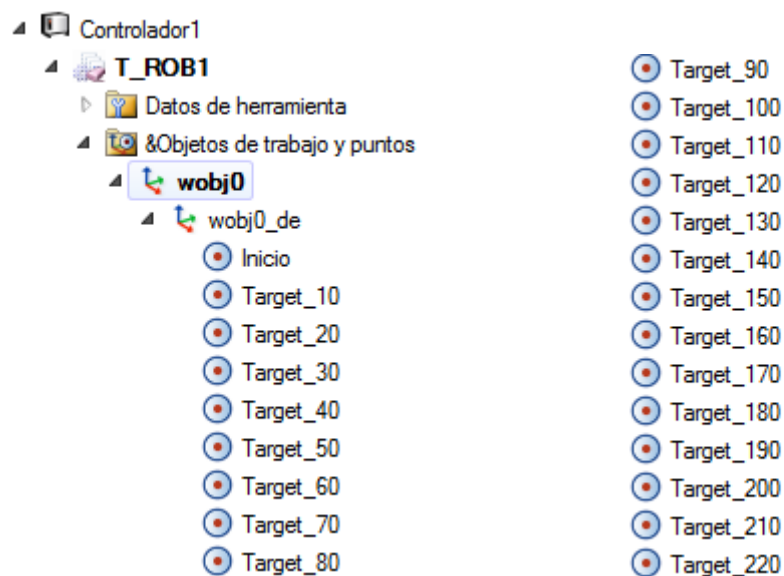
1. Cambiar la caja de una cinta a otra.
2. Realizar una trayectoria por los taladros de la pieza.



Para realizar todos estos movimientos he añadido un **controlador virtual** al robot. Una vez instalado el controlador, se crearan cuatro nuevas señales:

- **Dos de entrada:** que darán comienzo a la simulación.
- **Dos de salida:** que darán fin a la simulación.

Una vez añadido el controlador, añadiremos los puntos (que recorrerá el robot) al objeto de trabajo **wobj0**.



Dichos puntos serán los que recorra el robot por el path, en nuestro caso concreto el **path10**. Este path será el encargado de sincronizarse con RAPID y la simulación. Para ello, haremos que la caja se mueva con un **LinearMover** choque con el **PlaneSensor** que activara la entrada uno e inicializando las dos señales de salida a cero.

Llegados a este punto el robot realizará la trayectoria de los taladros, tanto circulares como cuadráticos. Para poder realizar estos movimientos es necesario saber qué tipo de movimiento se realizará. Para los puntos donde se conecte una línea recta (taladros cuadráticos) se usará el **MoveL** que hace referencia a un movimiento lineal. En cambio para los puntos donde se conecte una curva se usará el **MoveC** que hace referencia a un movimiento circular.

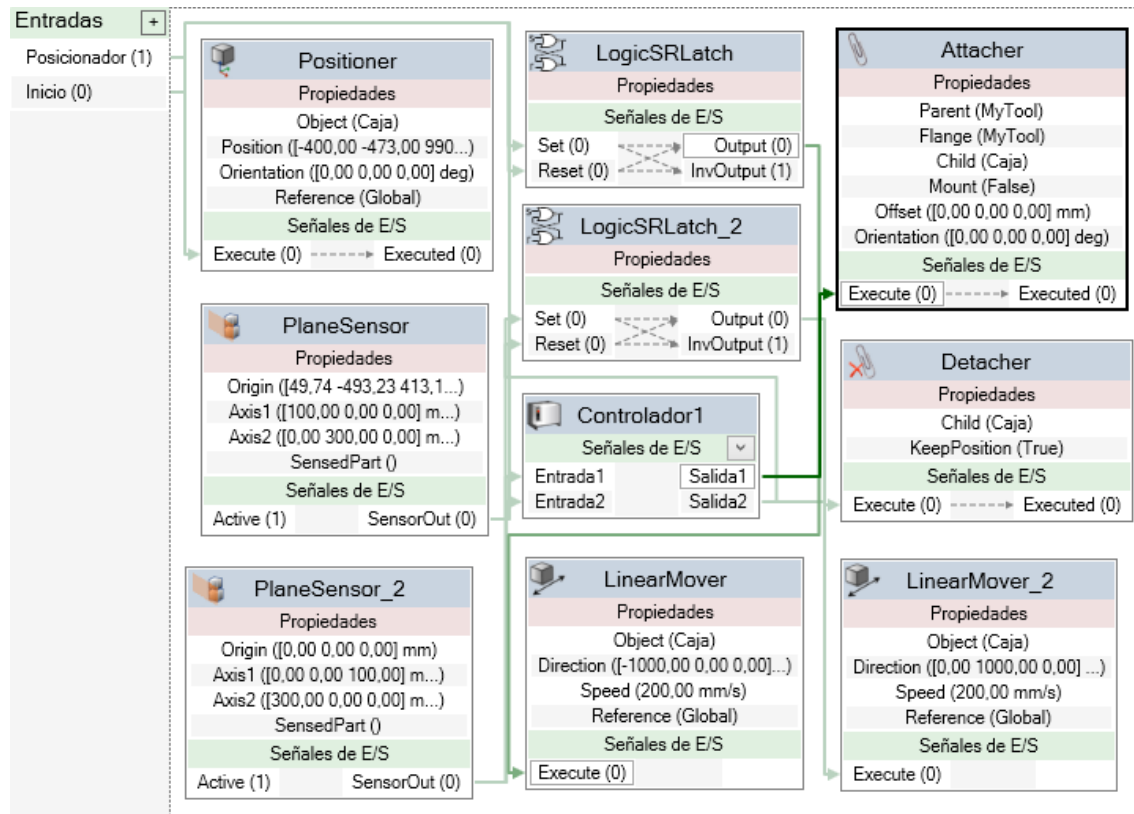
Para activar las señales de salida y que el robot mueva la caja de una cinta a otra, se utilizara el **MoveLdo** que se basa en mover la caja por un punto mientras inicializas una de las señales de salida a uno.

Por último, cuando la caja ya haya sido colocada en la cinta dos, haremos que la caja se vuelva mover con un **LinearMover** hasta otro **PlaneSensor** donde la entrada dos se pondrá a uno, esperando a que la caja choque con el **PlaneSensor**.



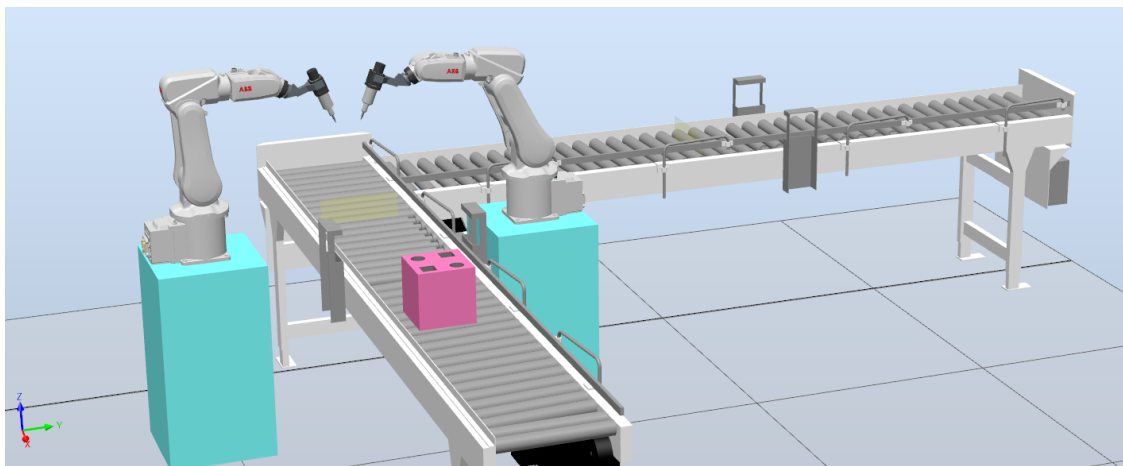
Para realizar todas las conexiones entre las entradas y salidas utilizaremos la **Lógica de Estación**. En ella uniremos cada salida y entrada con su correspondiente objeto. En este caso se hará uso de dos **PlaneSensor**, dos **LogicSRLatch**, dos **LinearMover**, un **Attacher**, un **Detacher**, un **SimulationEvents** y el **Controlador**.

También se puede añadir un **Positioner**, si se quiere, sirve para colocar la caja (en nuestro caso) en una determinada posición, esta señal se añade desde **señales de la estación** y se conectara con el posicionador (así cuando la caja se mueva del sitio en donde queremos que empiece y le demos al posicionador, este recolocara la caja automáticamente haciendo que la señal **posicionador** se ponga a uno).



PARTE 2: DOS CONTROLADOR Y DOS ROBOT

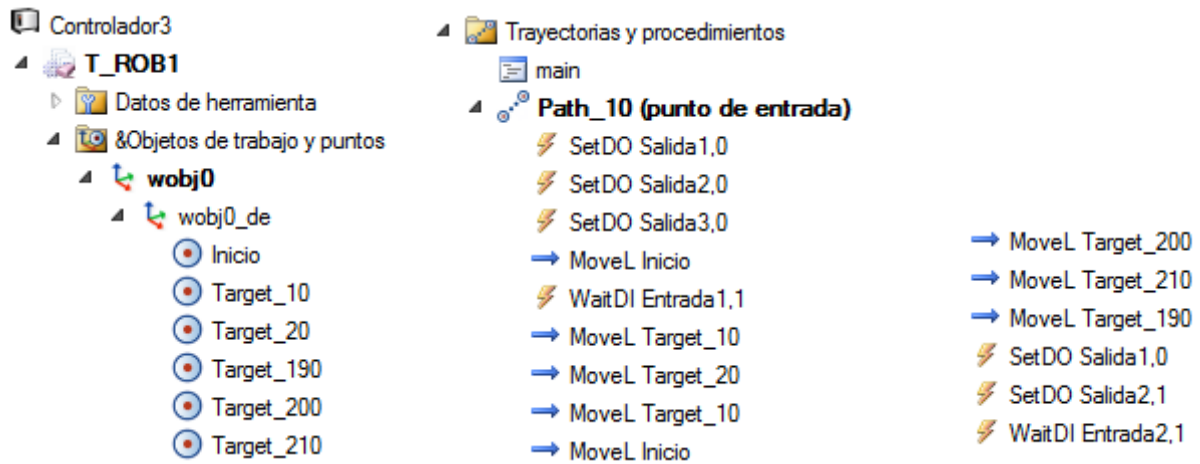
Para esta segunda parte escogeremos el mismo robot, el **robot IRB 120**, y lo colocaremos de forma que queden así. Además de que añadiremos un segundo controlador para controlar cuando empieza a moverse el segundo robot (en este caso es el robot de la derecha):



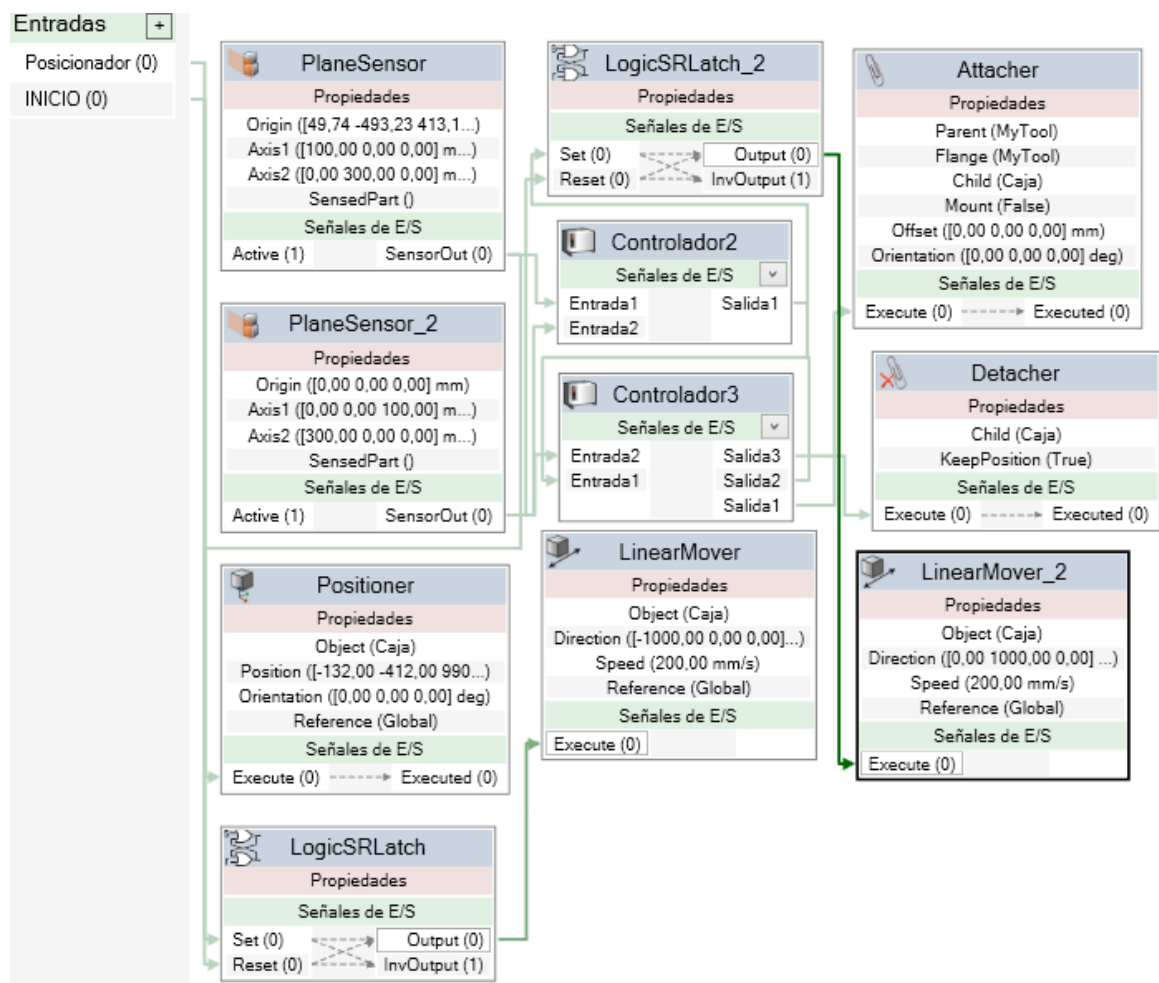
El robot de la izquierda se encargará de realizar las trayectoria por los taladros, es decir, la siguiente trayectoria:



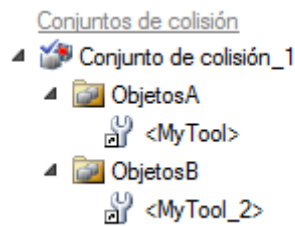
Una vez que este robot haya acabado su trayectoria la **salida uno** del **controlador dos** se pondrá a uno y le dirá al **controlador tres** que el robot de la derecha puede empezar su trayectoria, que es la siguiente:



Quedando como resultado una lógica de estación tal que así:



Para la gestión de colisiones lo único que haremos es **crear un conjunto de colisiones**, en la pestaña de **simulación**. Y añadiremos cada **MyTool** a cada objeto, es decir, la herramienta del robot de la izquierda ira al **objeto A** y la herramienta del robot de la derecha ira al **objeto B**. Obteniendo lo siguiente:



OBSERVACIONES: PARA AMBAS PARTES

Para establecer la simulación de ambas partes del trabajo lo que he hecho, ha sido añadir una señal a **la señal de estación (inicio)** para simbolizar el inicio de la simulación. Es decir, aunque le des a simular la caja no se moverá hasta que la señal de entrada (*inicio*) este en uno. Es decir, pulsando en la señal **inicio** la simulación deberá comenzar.

Para restablecer la simulación habrá quedarle al botón **Primer**, en vez de a restablecer, ya que este guarda los parámetros todos a 0. Es decir, para que la caja se mueva hacia el eje **x** y siempre se mueva sobre el eje **y**, al iniciar la simulación. Para qué choque con el primer PlaneSensor y la simulación siga su curso.