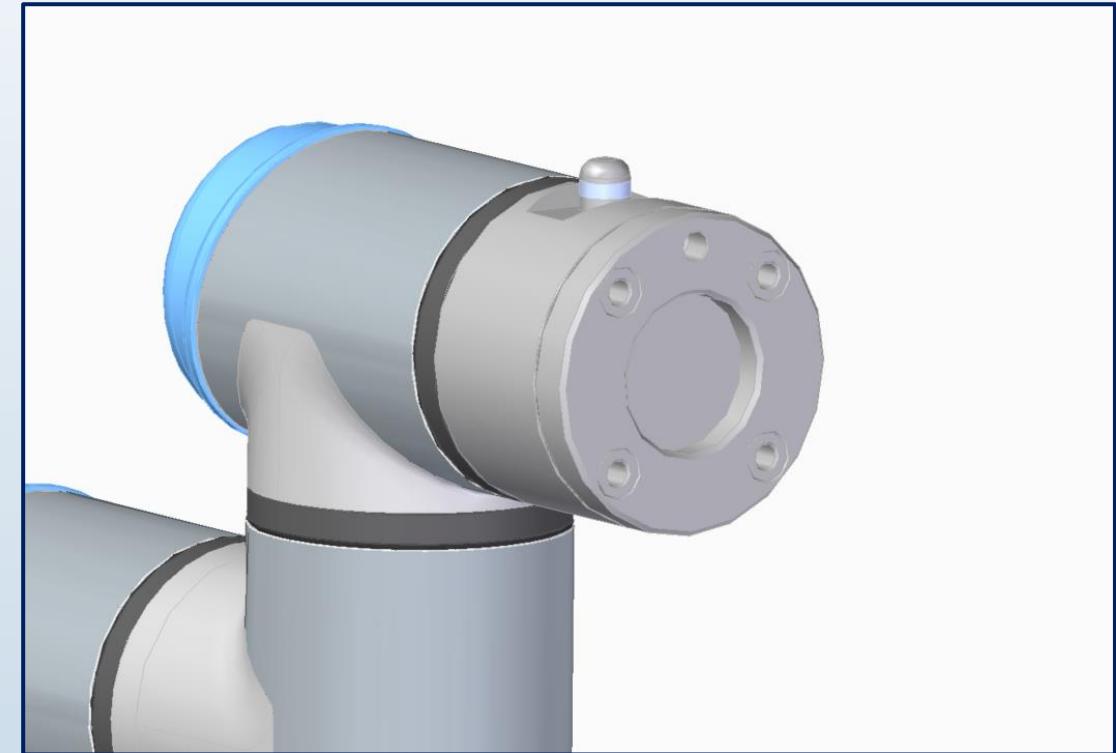
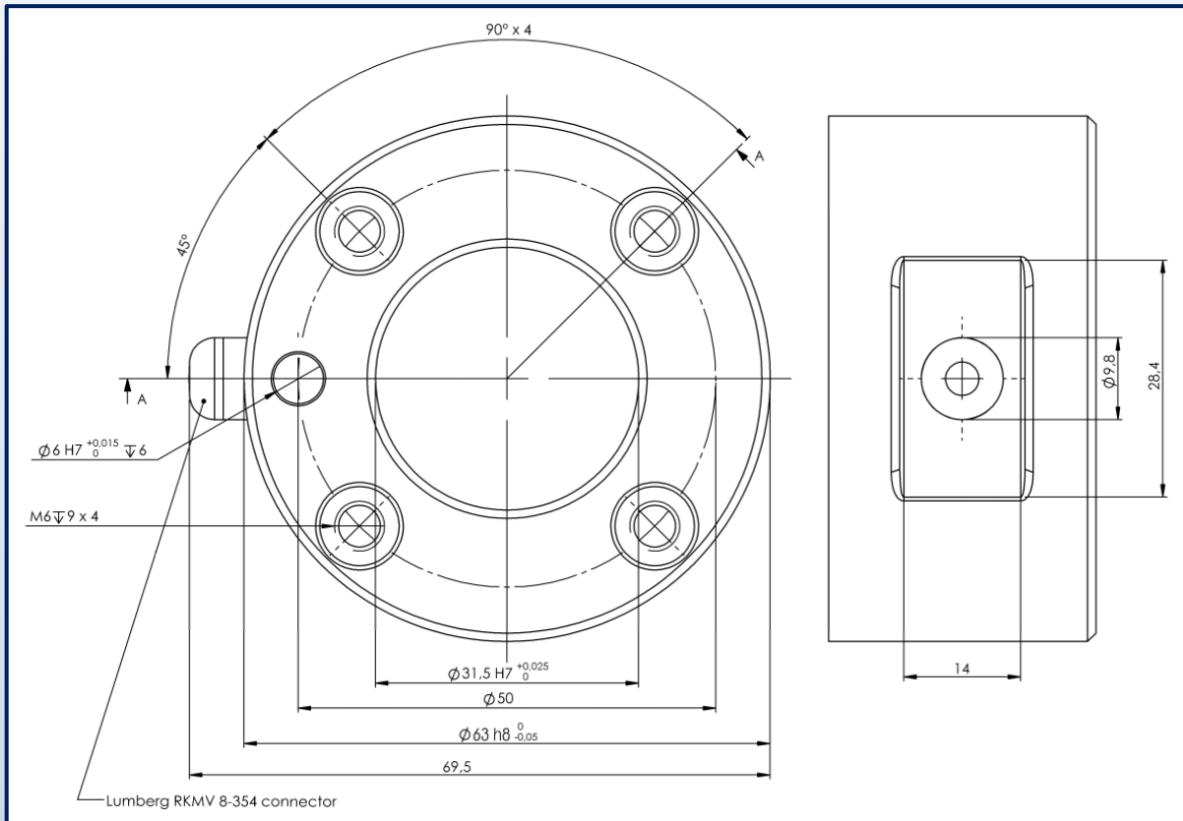


Robot colaborativo UR3

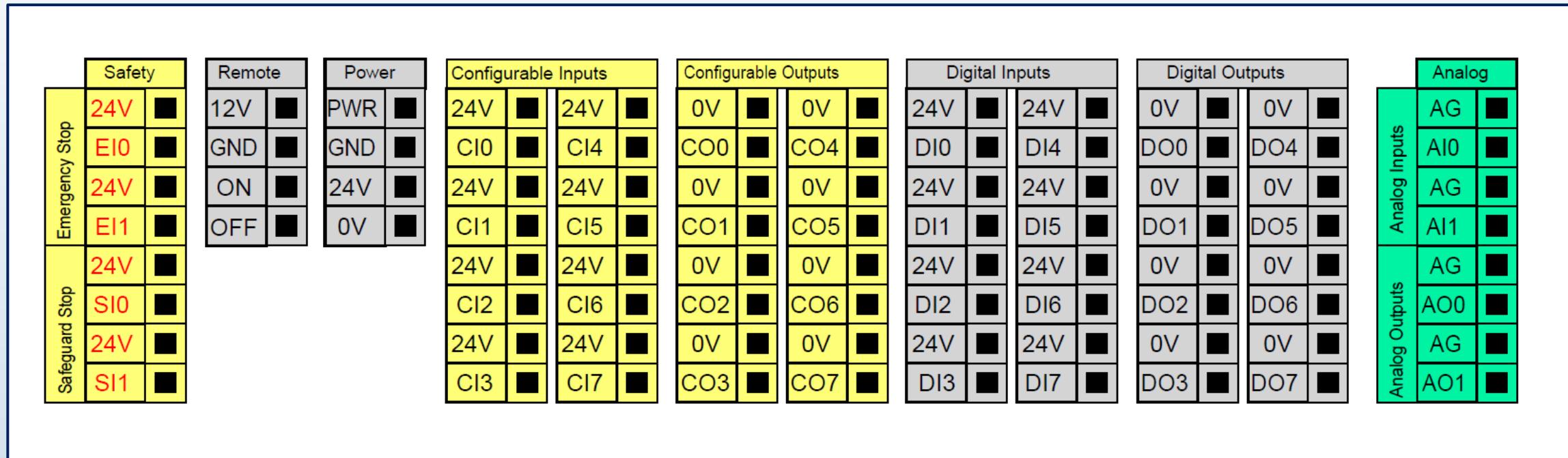
Pinza neumática



Cabezal



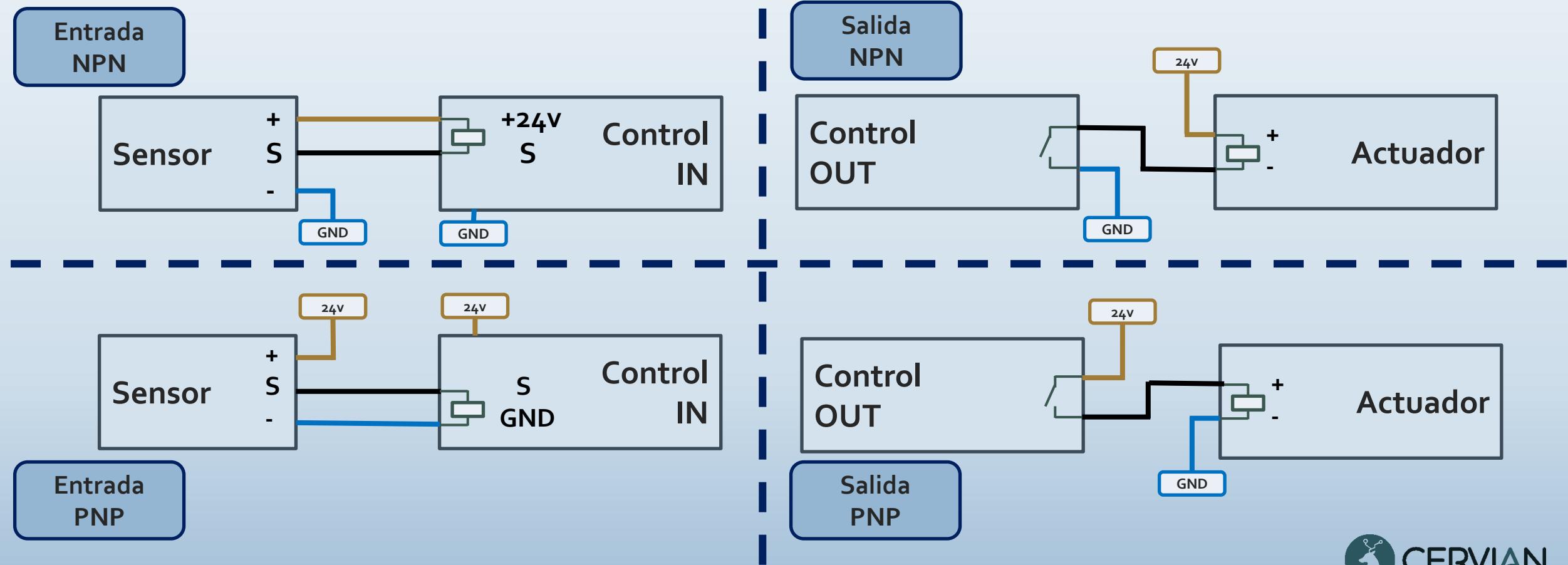
E/S de controlador



Color	Significado
Amarillo con texto rojo	Señales de seguridad dedicadas
Amarillo con texto negro	Configurable para seguridad
Gris con texto negro	E/S digital de uso general
Verde con texto negro	E/S analógica de uso general

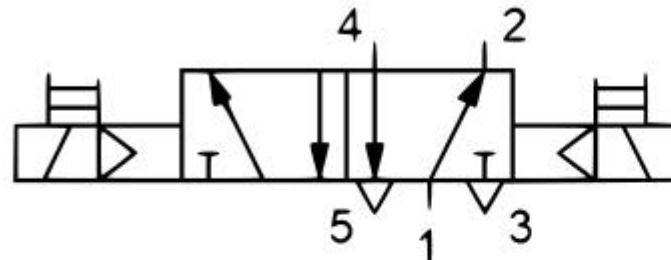
E/S de controlador

Entradas	Salidas
Tipo PNP	Tipo PNP
La señal es 1 cuando recibe 24V	Cuando se activa proporciona 0V

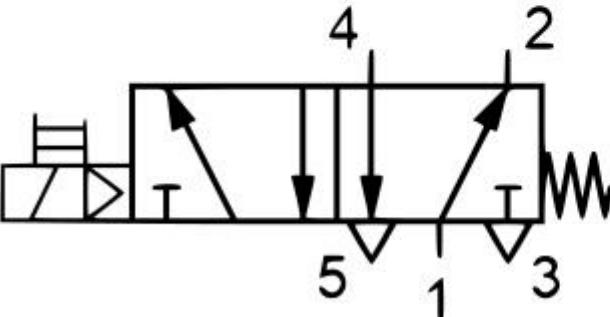


Válvula 5/2

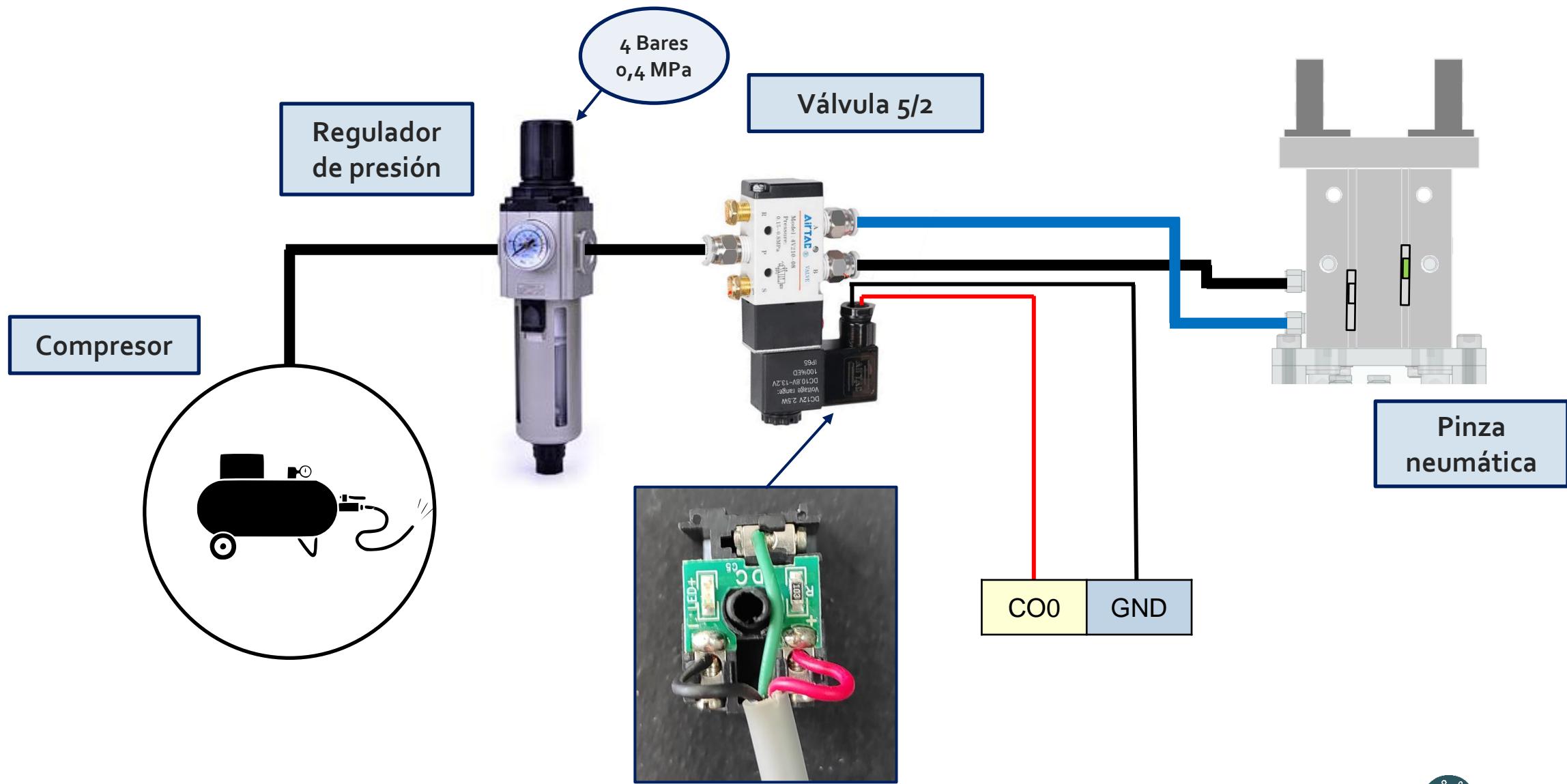
Electro válvula
biestable



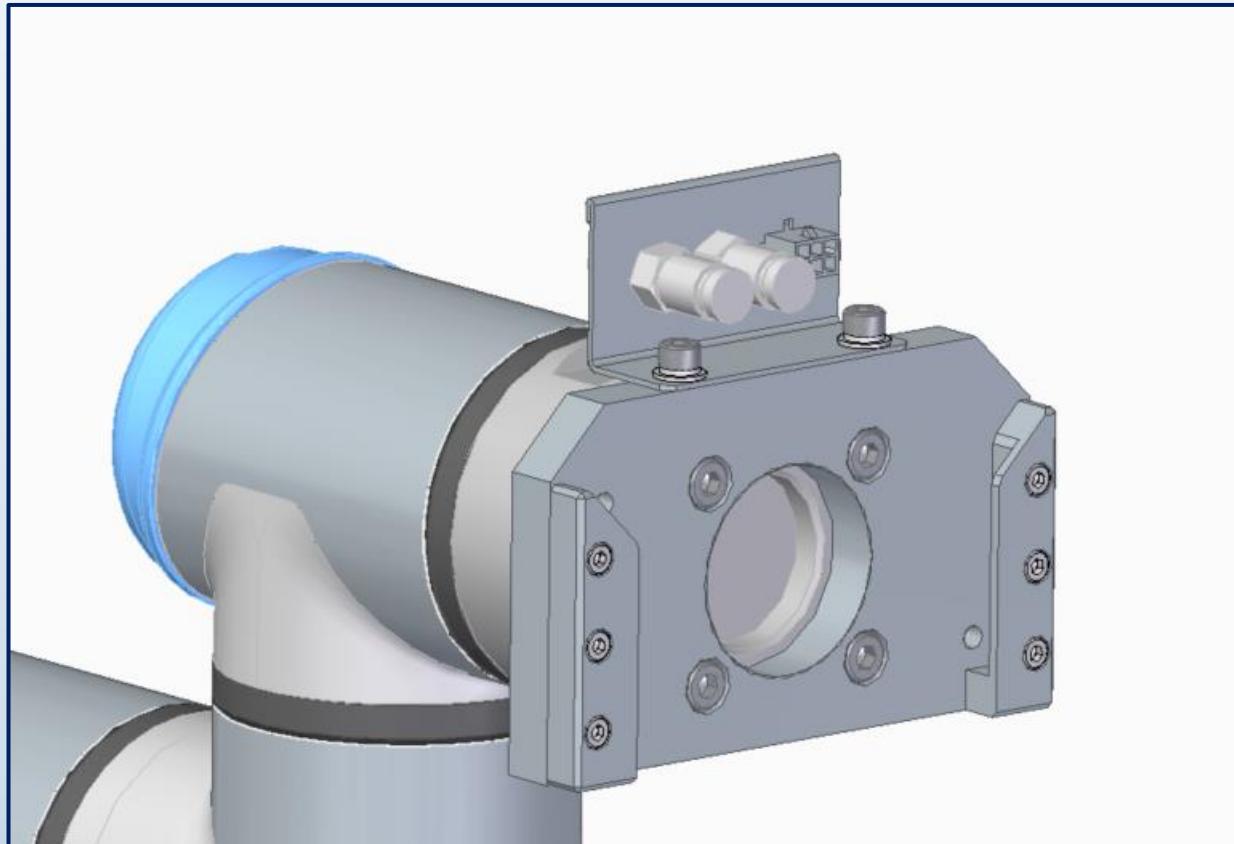
Electro válvula
monoestable



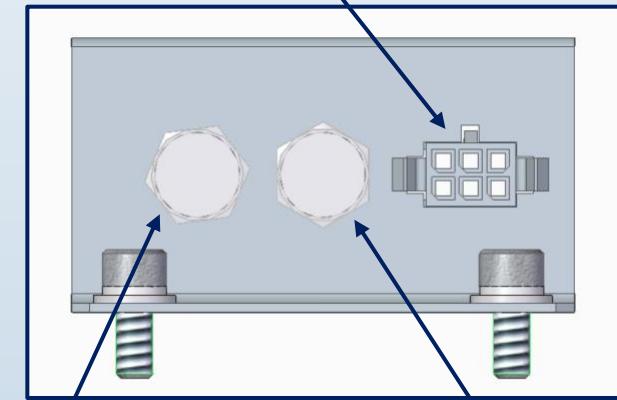
Circuito neumático



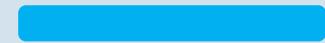
Base del cabezal



Conector Molex hembra				
Nº pin	6	5	4	Sensor 2
Tipo	GND	+24V	Señal	
Color	Gris	Rojo	Blanco	
Nº pin	3	2	1	Sensor 1
Tipo	GND	+24V	Señal	
Color	Azul	Marrón	Negro	



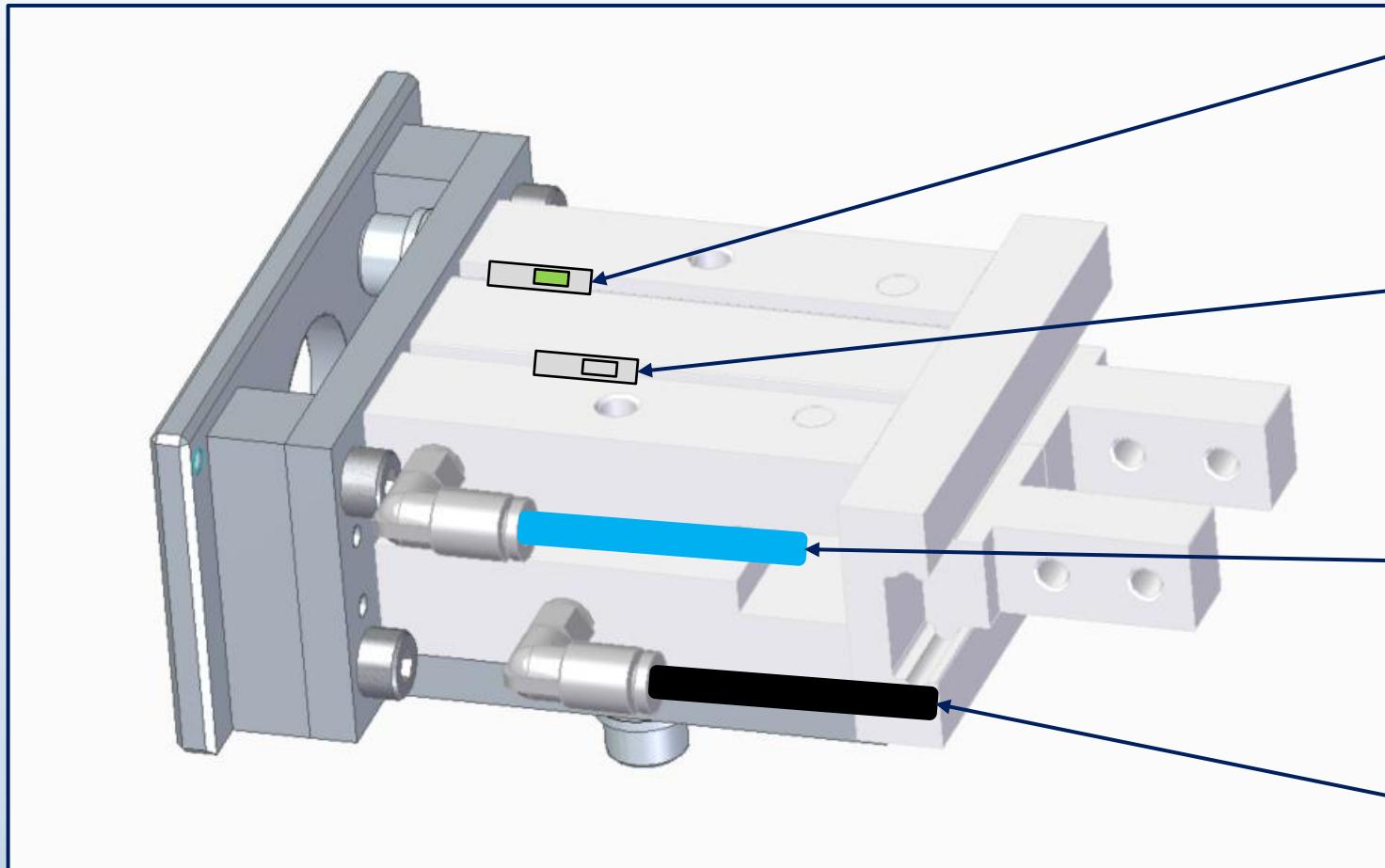
Neumático A



Neumático B



Conexiones del cabezal



Sensor 2

FC1

Detección de cierre
de pinza

Sensor 1

FC0

Detección de
apertura de pinza

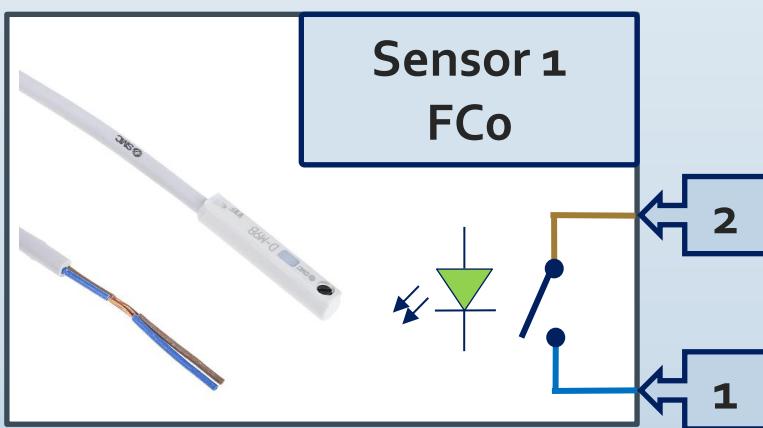
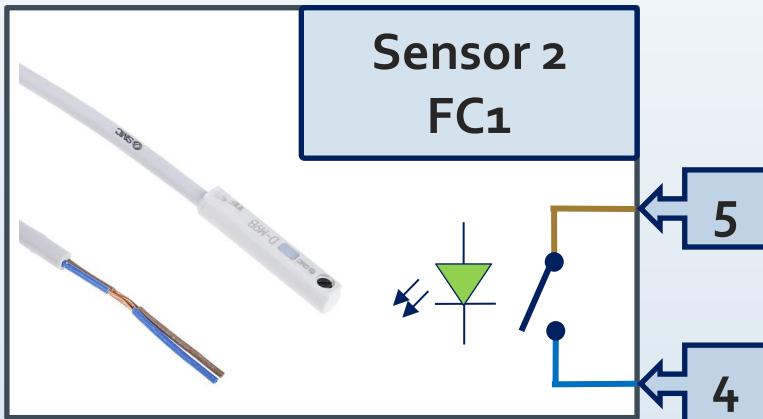
**Conexión
neumática A**

Aporte de aire en
reposo

**Conexión
neumática B**

Aporte de aire en
activo

Conexión de finales de carrera



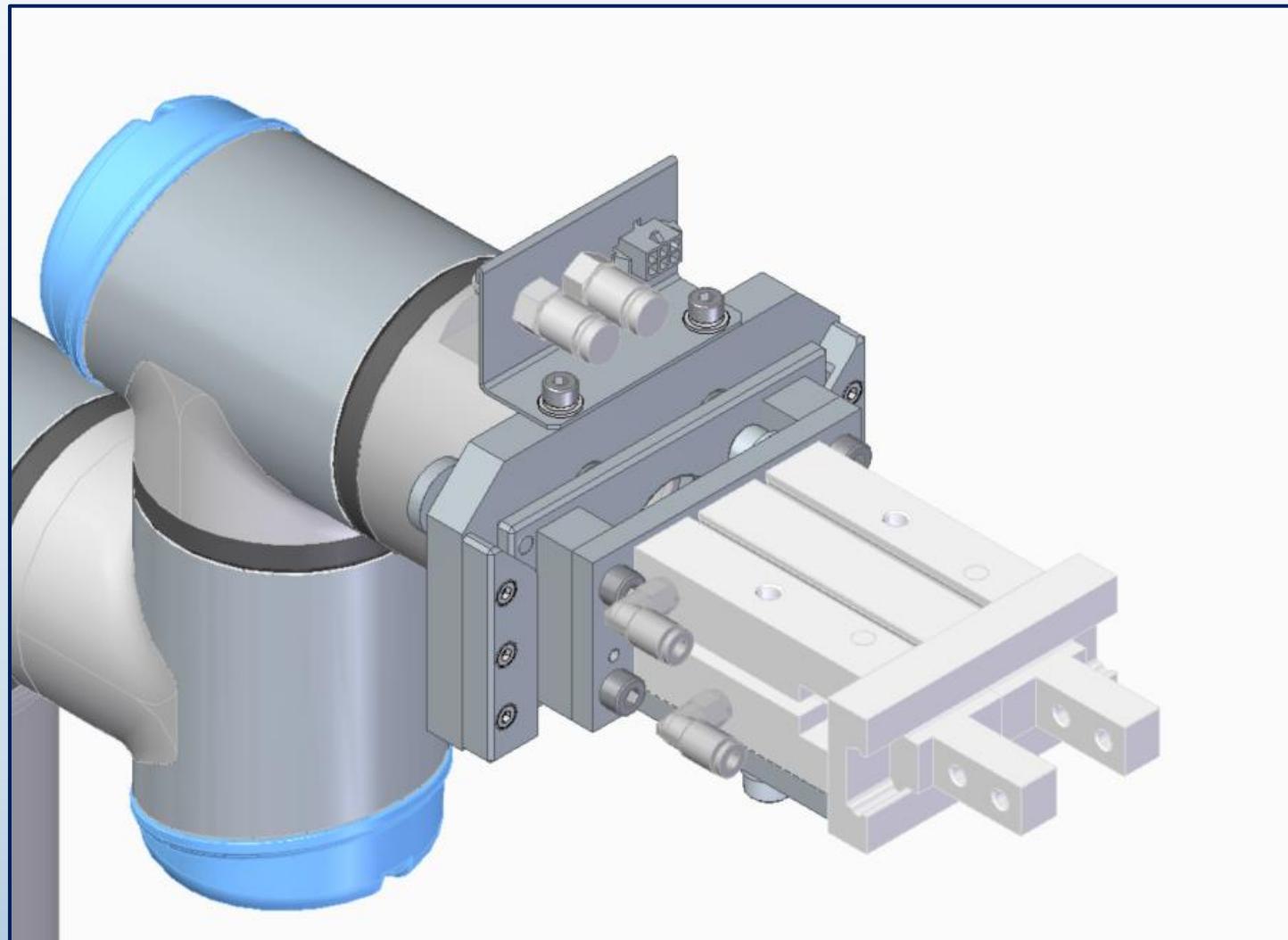
Conector Molex 6 vías macho			Controladora
6 GND	5 +24V	4 Señal	Cl1
3 GND	2 +24V	1 Señal	Clo

Cabezal de pinza

Instalación

Para instalar el cabezal de pinza:

1. Insertar el conjunto en las ranuras del cabezal.
2. Atornillar desde atrás con los tornillos M4. **No apretar con herramientas**
3. Conectar los tubos neumáticos donde corresponde.
4. Conectar los sensores de final de carrera.



Sensor de proximidad

El detector de proximidad es un sensor tipo infrarrojos de 24V tipo PNP con distancia de detección ajustable de 3 a 80 centímetros.

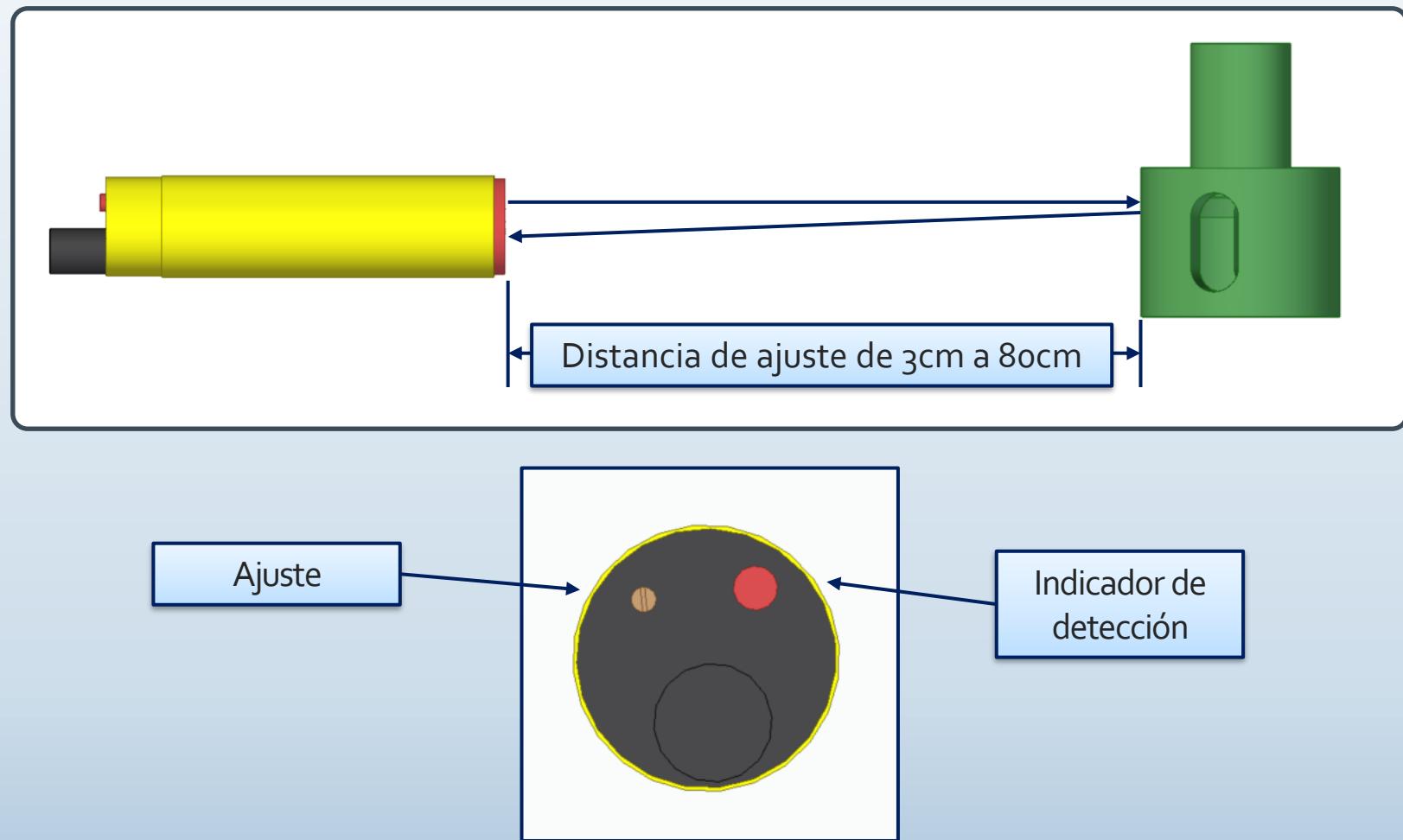


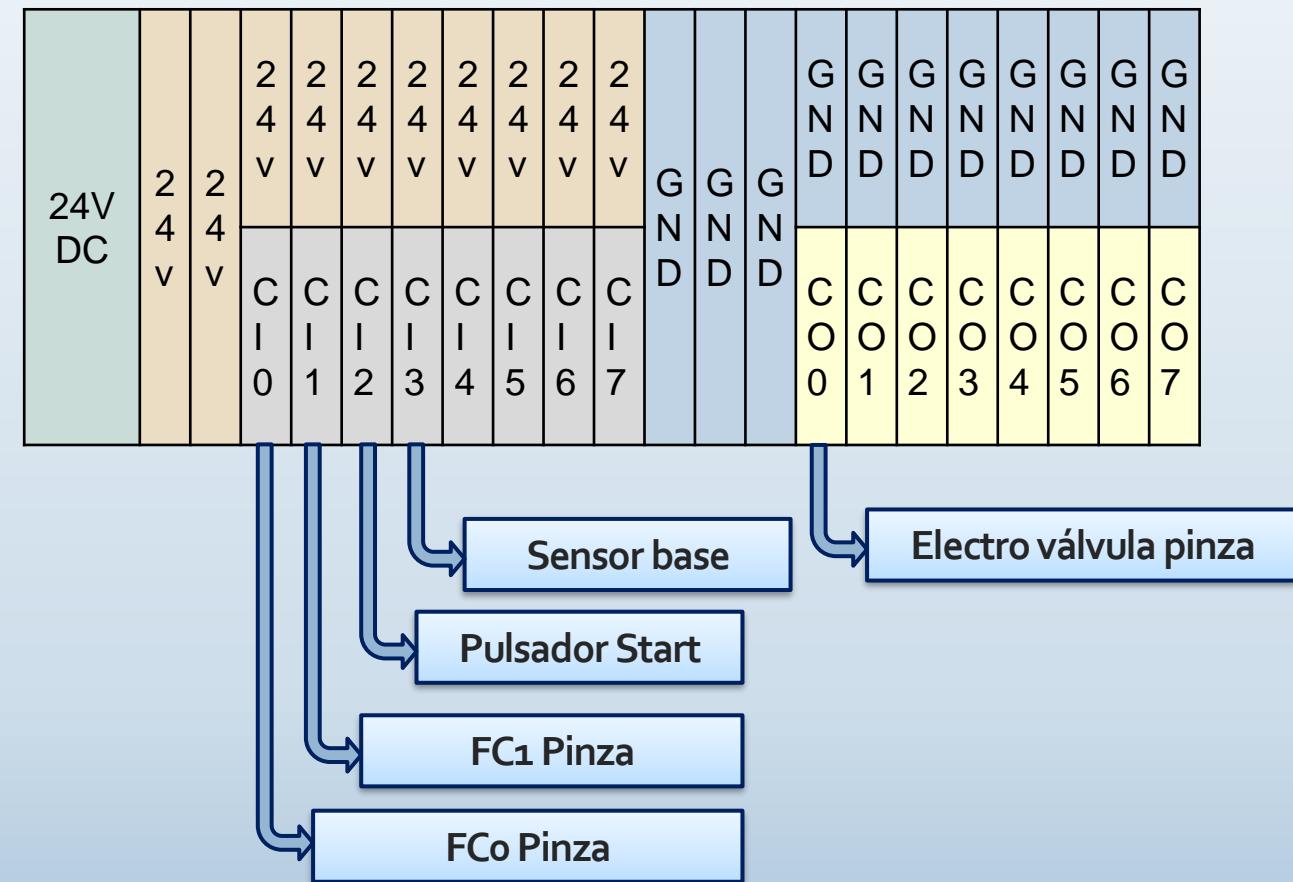
Diagrama de conexiones

Entradas:

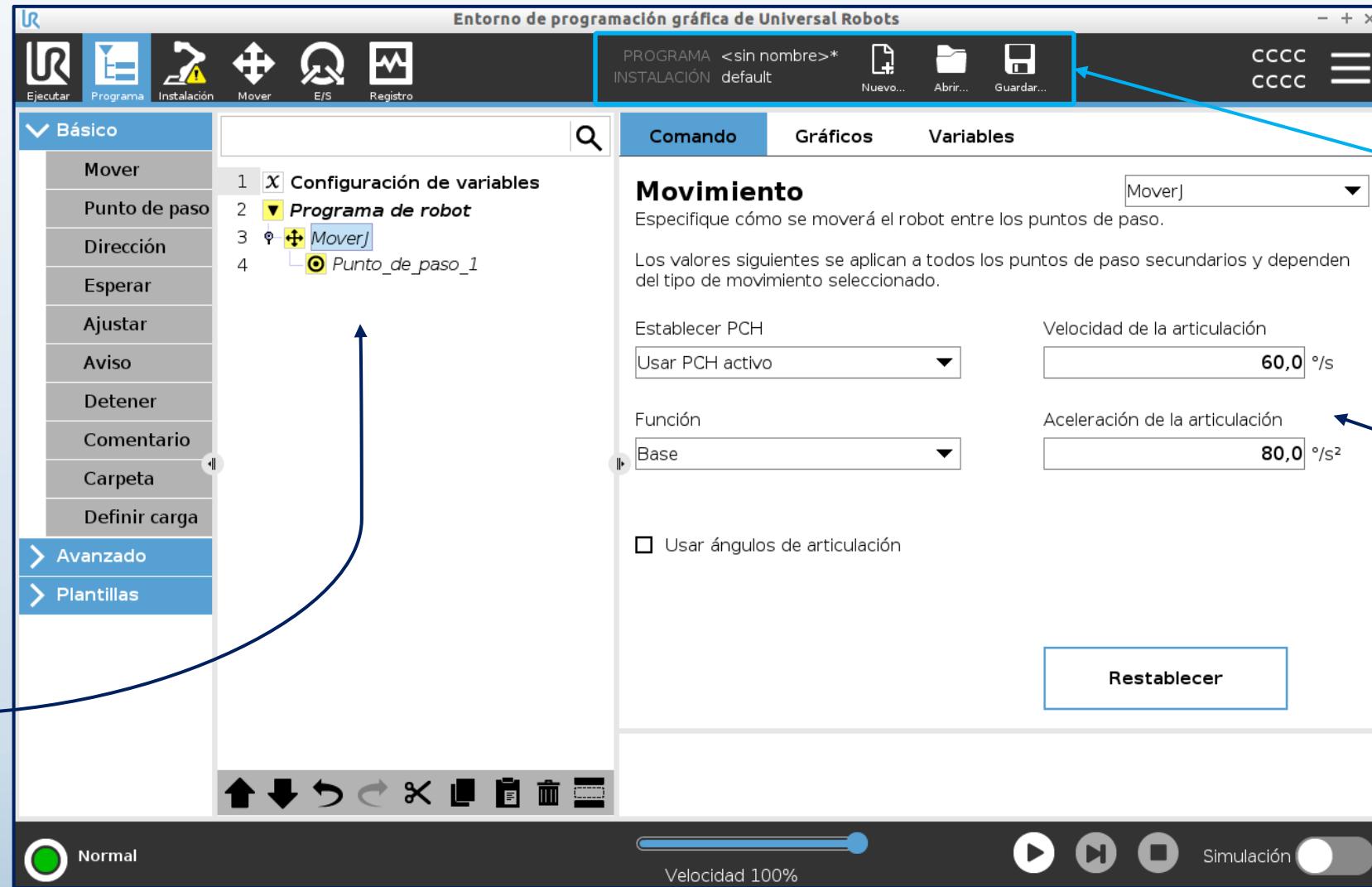
- Clo: FCo (final de carrera en pinza abierta).
 - Cl1: FC1 (final de carrera de pinza cerrada). Este sensor se ajustará en función al tamaño de la pieza a coger).
 - Cl2: Pulsador de inicio de secuencia.
 - Cl3: Sensor infrarrojos de presencia de la base.

Salidas:

- COo: Accionamiento de la electroválvula para cerrar pinza.



Creación de nuevo programa



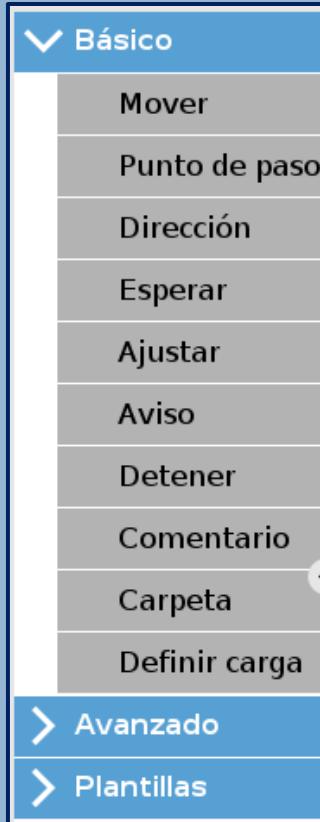
Comandos de programación

Crear, abrir o guardar programa

Configuraciones de comandos

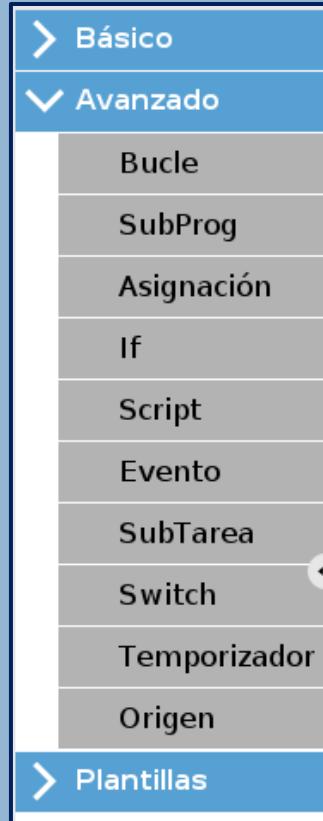
Secuencia de programa

Comandos básicos



- Mover: Controla el movimiento del robot. Se compone de puntos de paso. Existen tres tipos de movimientos:
 - I. MoveJ: Realiza la trayectoria ajustando los ejes individualmente y a la misma velocidad. Crea trayectorias curvas.
 - II. MoveL: Hace que la herramienta se mueva linealmente.
 - III. MoveP: Movimiento lineal a velocidad constante. Se suele usar en dispensaciones.
- Punto de paso: Puntos de la trayectoria del robot.
- Dirección: Configuración de los tipos de movimiento MoveL o MoveP
- Esperar: Espera del programa por tiempo o por condición (sensor externo).
- Ajustar: Actúa sobre las señales de salida digitales o analógicas.
- Aviso: Configura mensajes en pantalla cuando llegue a este punto.
- Detener: Detección del programa.
- Comentario: Añadir comentario en una línea de código.
- Carpeta: Sirve para organizar el programa.
- Definir carga: Redefine la carga de la herramienta. Se usa cuando se coge o se suelta un objeto.

Comandos avanzados



- Bucle: Realiza bucles de programa que se repiten hasta el infinito, un número determinado de veces o siempre que la condición dada sea verdadera.
- SubPrograma: Creación de sub rutinas que se pueden usar varias veces en el programa.
- Dirección: Configuración de los tipos de movimiento MoverL o MoverP
- Asignación: Sirve para asignar valores a las variables.
- If: Condicionante de estructura de control para realizar tareas dependientes de una o varias señales. Si la condición es "verdad", se ejecuta las líneas incluidas dentro del bloque.
- Script: Comando para ejecución de subrutinas externas (Web).
- Evento: Funciona como interrupción del programa cuando se cumple la condición. .
- SubTarea: Permite ejecutar una secuencia de programa en paralelo al principal.
- Switch: Sirve para controlar varias posibilidades en función de un valor de una variable.
- Temporizador: Se puede iniciar, detener o restablecer.
- Origen. Sirve para establecer un origen de trabajo

Programar entrada

Podemos usar una señal de entrada de diferentes formas. En este ejemplo, usamos una señal de condicionante (1) o esperamos a que esa señal se produzca (2).

El comando "If" se encuentra en las herramientas avanzadas y el "Esperar" en herramientas básicas.

The image shows a software interface for programming a robot. The left side is a code editor with a tree view of the program structure. The right side is a configuration panel for the 'Esperar' (Wait) command.

Code Editor (Left):

- Line 4: Ajustar DO[0]=Apagar
- Line 5: **Programa de robot**
- Line 6: MoverJ
- Line 7: Punto_Inicial
- Line 8: **If digital_in[2]≠ True**
- Line 9: MoverL
- Line 10: Punto_1
- Line 11: Punto_aprox_1
- Line 12: Esperar: 1.0
- Line 13: Ajustar DO[0]=Encender
- Line 14: Esperar: 1.0
- Line 15: **Esperar DI[1]=HI** (highlighted in blue)
- Line 16: Esperar: 1.0
- Line 17: Ajustar DO[0]=Apagar
- Line 18: Esperar: 1.0
- Line 19: Esperar DI[0]=HI
- Line 20: Punto_1
- Line 21: MoverL
- Line 22: MoverL
- Line 23: MoverL
- Line 24: MoverL
- Line 25: MoverJ

Configuration Panel (Right):

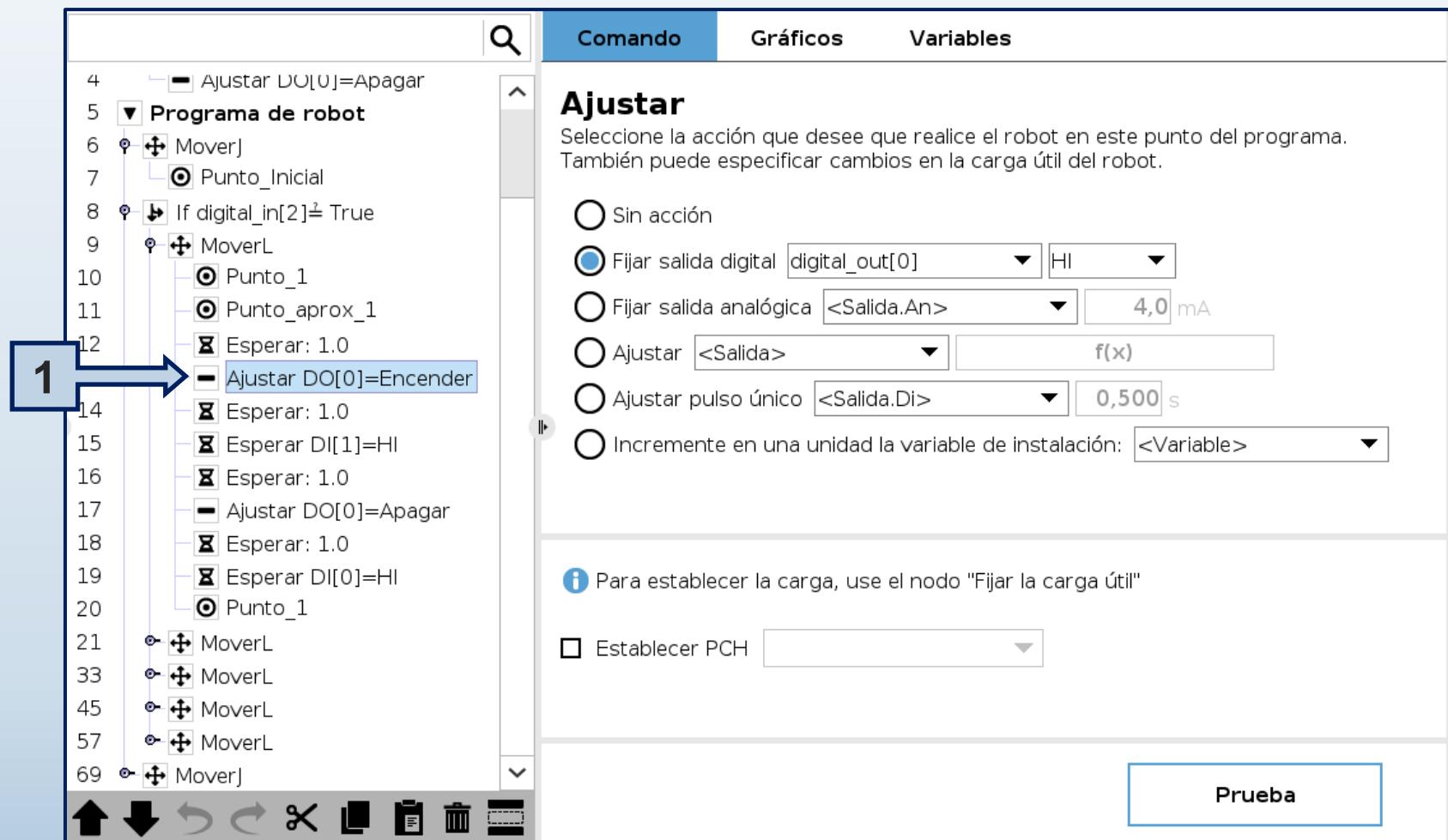
Esperar
Seleccione qué debe activar la siguiente acción del robot;

- No esperar
- Esperar **0,01** segundos
- Esperar entrada digital **digital_in[1]** **HI**
- Esperar **<Entrada.An>** **>** **4,0** mA
- Esperar **f(x)**

Programar salida

Para activar una salida, usamos el comando “ajustar” y ponemos la señal digital en “HI” (1). Para apagar, el mismo comando en “LO”.

El comando “Ajusta” se encuentra en las herramientas básicas.



The screenshot shows a software interface for programming a robot. The left side displays a list of commands in a tree structure:

- 4 Ajustar DO[0]=Apagar
- 5 Programa de robot
- 6 MoverJ
- 7 Punto_Inicial
- 8 If digital_in[2] != True
- 9 MoverL
- 10 Punto_1
- 11 Punto_aprox_1
- 12 Esperar: 1.0
- 13 Ajustar DO[0]=Encender **1**
- 14 Esperar: 1.0
- 15 Esperar DI[1]=HI
- 16 Esperar: 1.0
- 17 Ajustar DO[0]=Apagar
- 18 Esperar: 1.0
- 19 Esperar DI[0]=HI
- 20 Punto_1
- 21 MoverL
- 33 MoverL
- 45 MoverL
- 57 MoverL
- 69 MoverJ

The command at index 13 is highlighted with a blue box and a large number '1' inside. A blue arrow points from this highlighted command to the configuration panel on the right.

Ajustar

Seleccione la acción que desea que realice el robot en este punto del programa. También puede especificar cambios en la carga útil del robot.

Sin acción

Fijar salida digital

Fijar salida analógica mA

Ajustar

Ajustar pulso único s

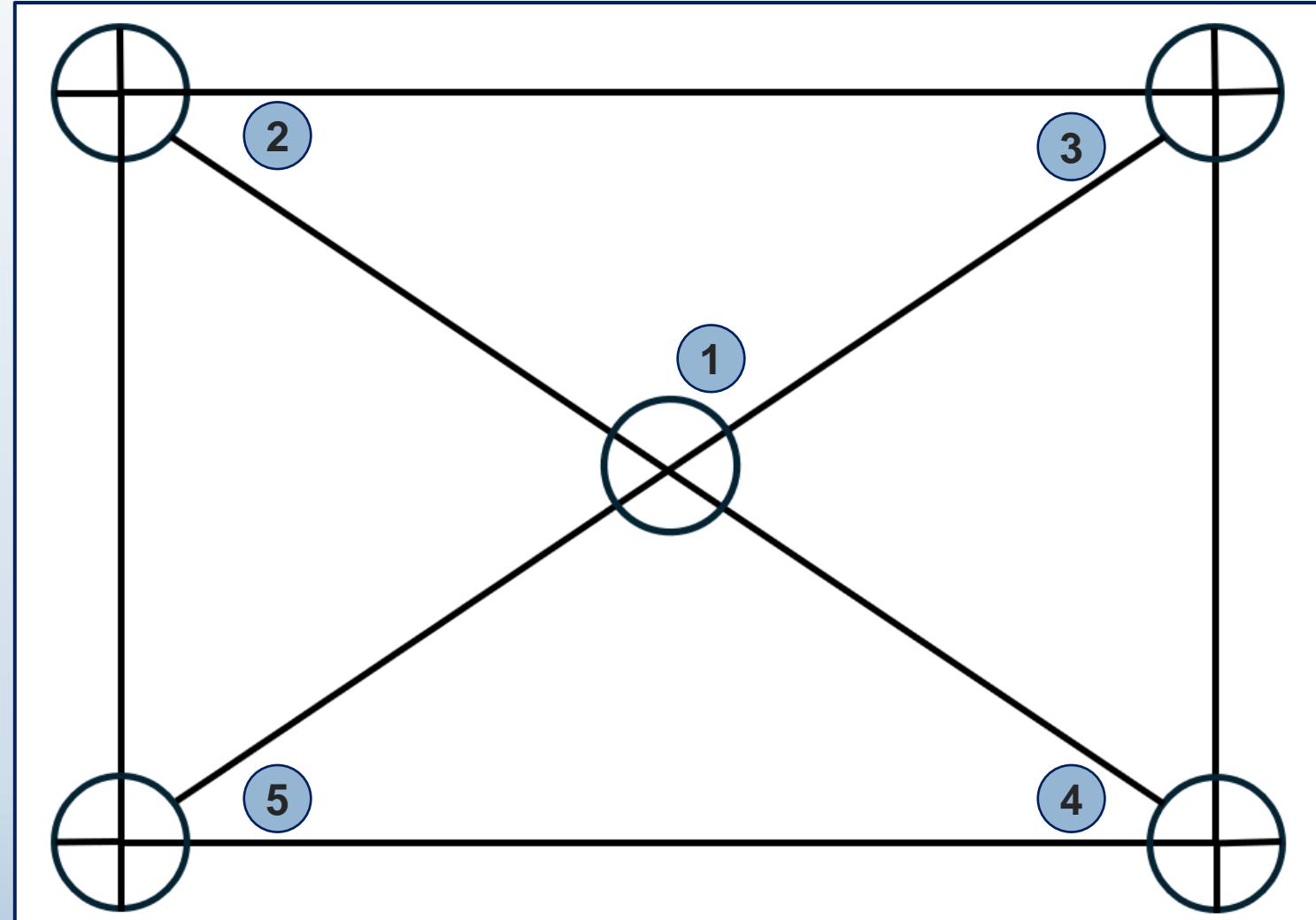
Incremente en una unidad la variable de instalación:

Prueba

Práctica 5. Movimientos de apertura y cierre

Con la herramienta de pinza, crear un programa para llevar la herramienta a los 5 puntos indicados partiendo desde origen tras pulsar Start. En cada punto, ha de cerrarse la pinza, chequear si se ha cerrado y abrirse de nuevo.

En cada punto, ha de realizarse aproximaciones en $-Z$ sobre la base teniendo la pinza en horizontal. Al finalizar, llevar a origen de nuevo.

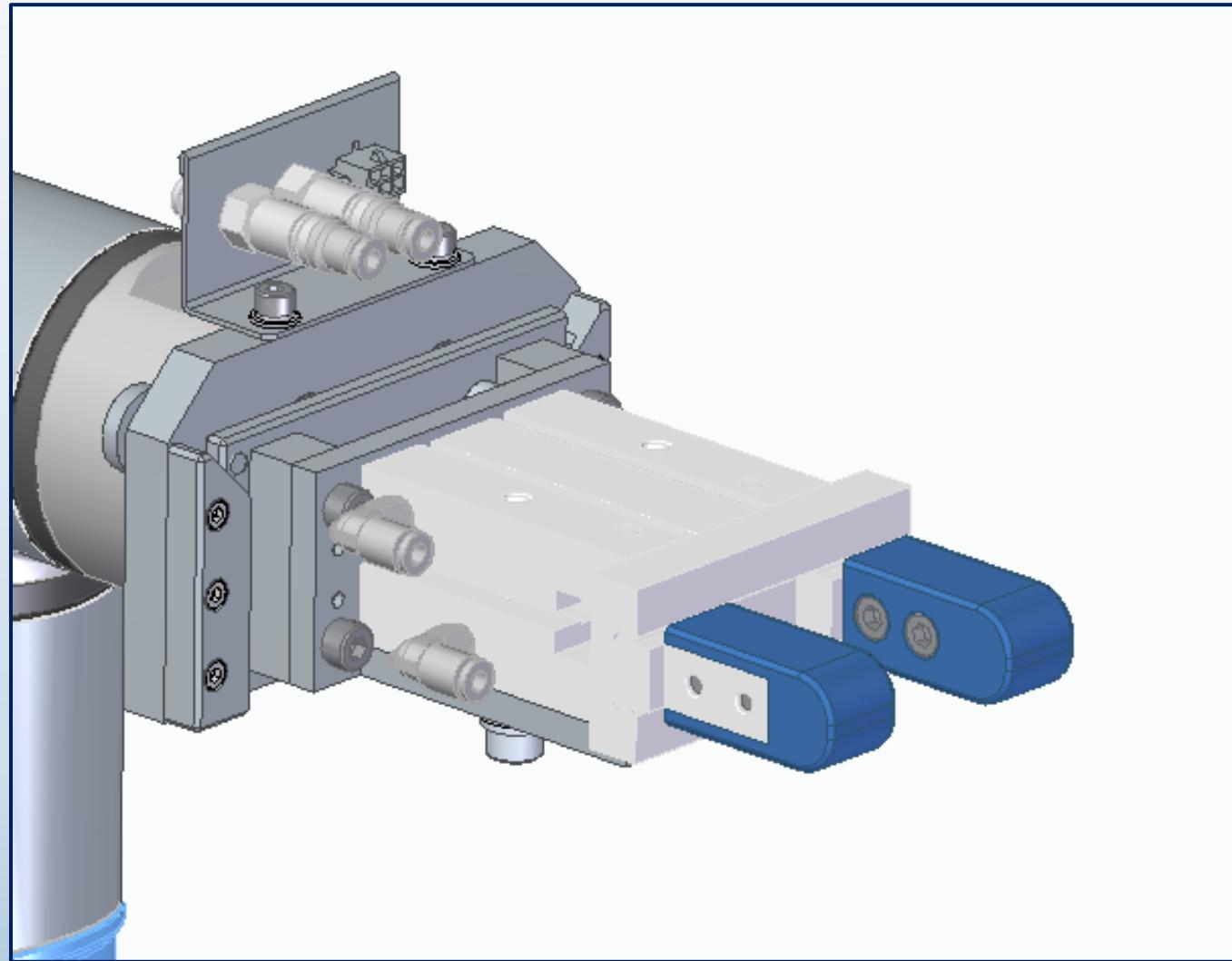


Pinza con dedos planos

Instalación

Los dedos de la pinza tienen dos agujeros por lado que sirven para adaptar cogidas en función de los objetos a coger.

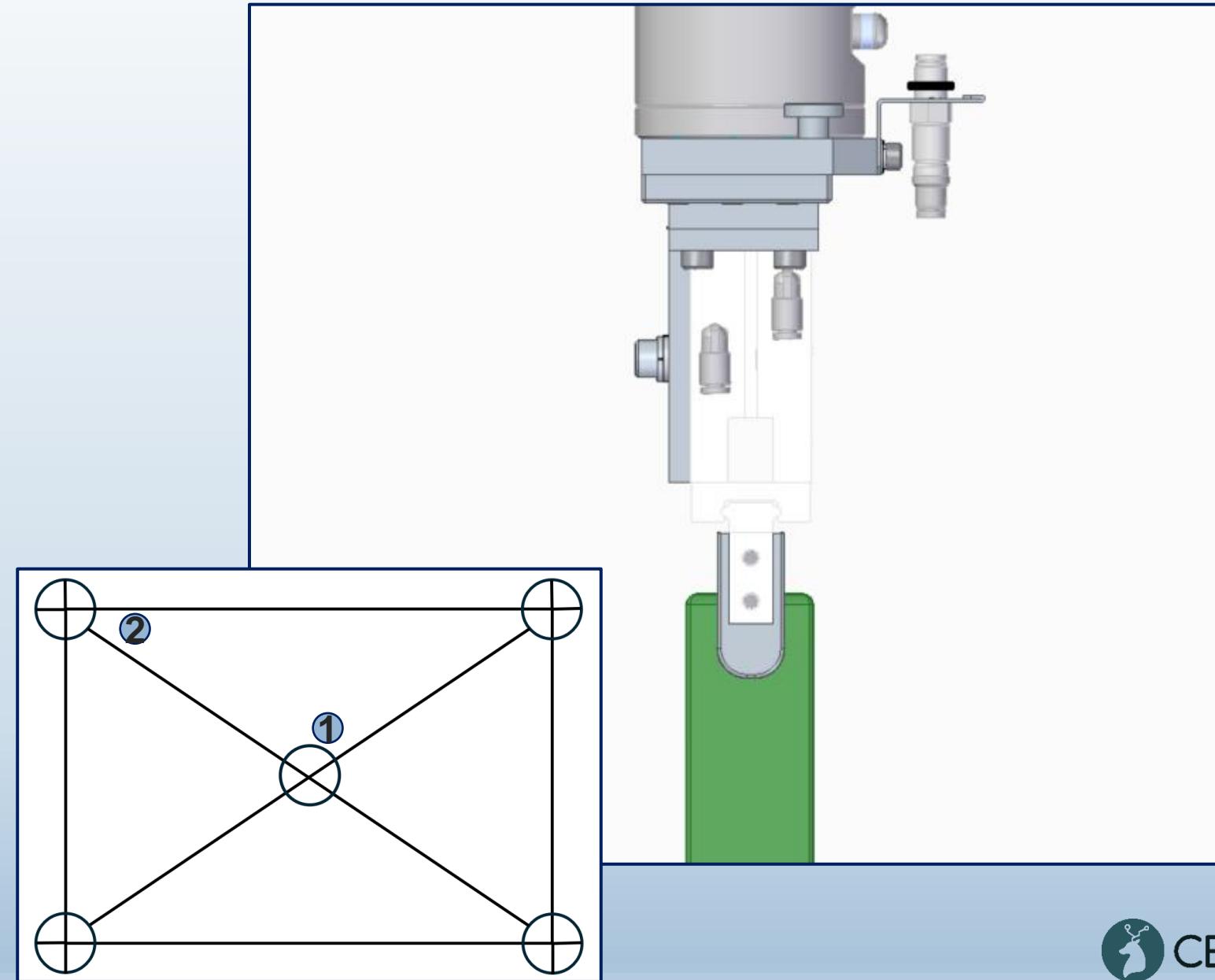
Para las primeras prácticas, instalaremos dos dedos simples con cara plana impresos en 3D.



Práctica 6. Desplazamiento de objetos

Con la herramienta de pinza, crear un programa para coger el objeto en punto 1 y soltarlo en el punto 2 partiendo desde origen tras pulsar Start. Una vez soltado el objeto, volver a origen y esperar 5 segundos. Por último, volver a por él al punto 2 y dejarlo en el punto 1 de nuevo y acabar en origen.

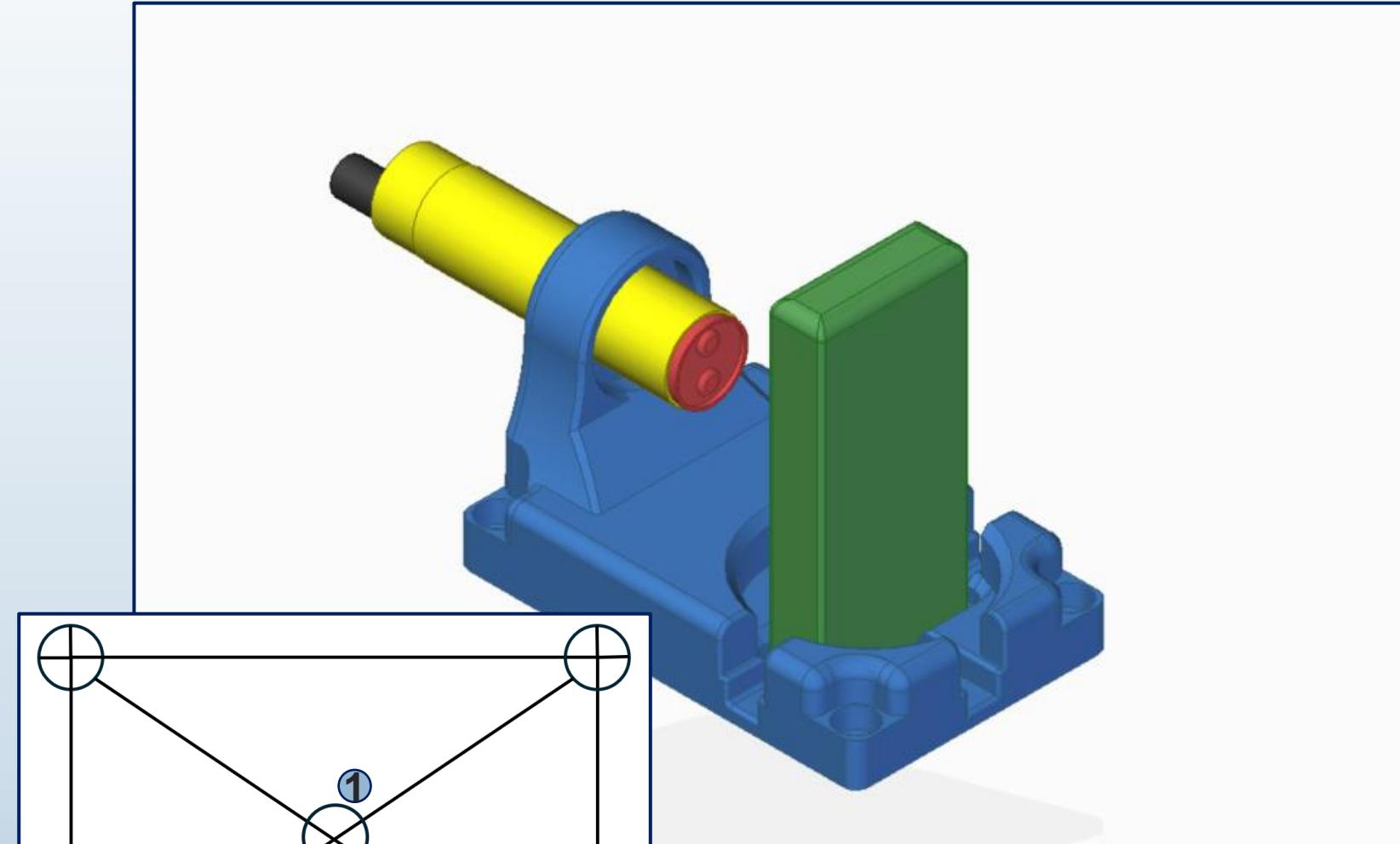
Las operaciones de cogida y suelta del objeto han de realizarse con aproximaciones en Z de base y con comprobaciones de los finales de carrera.



Práctica 7. Recogida en base

Con la herramienta de pinza, crear un programa para coger el objeto en la base (si se está detectando) tras pulsar Start. Llevar la pieza al punto 1 y volver a origen. Se ha de chequear la correcta cogida del objeto. Ejecutar el programa en bucle.

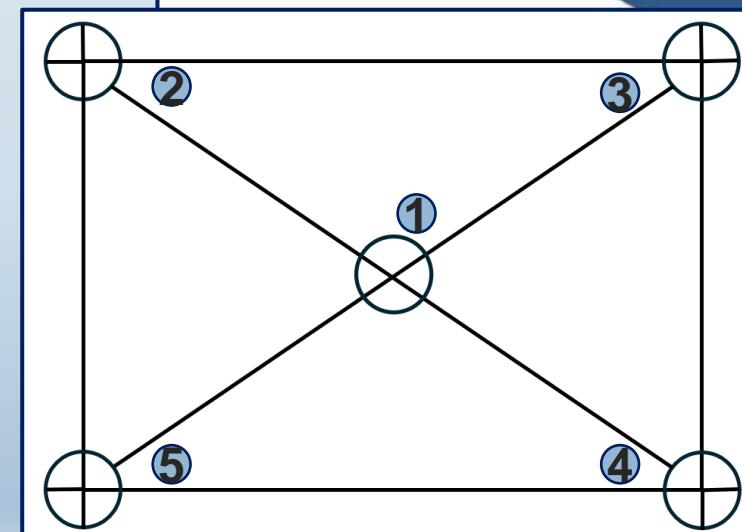
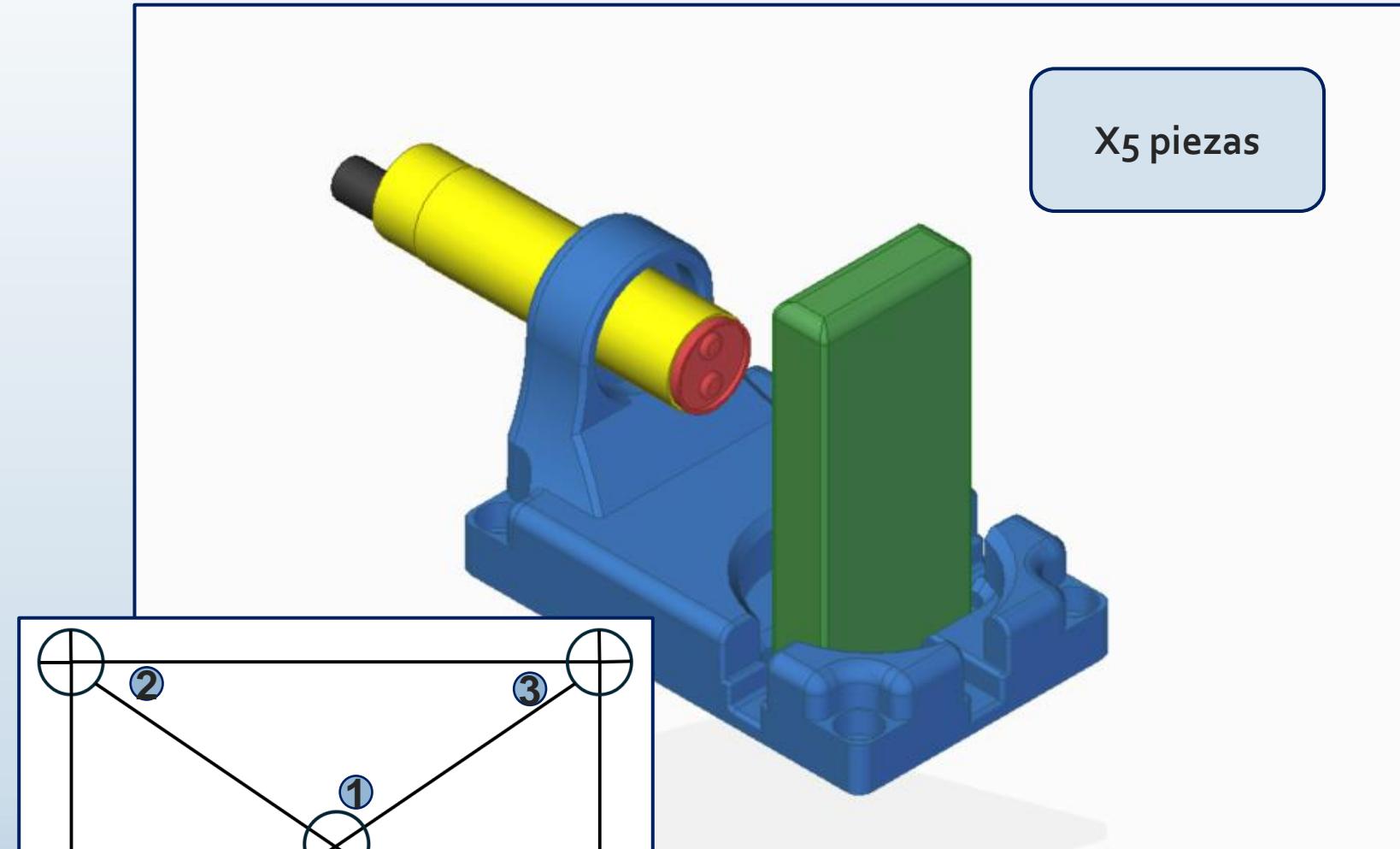
Las operaciones de cogida y suelta del objeto han de realizarse con aproximaciones en Z de base y con comprobaciones de los finales de carrera.



Práctica 8. Colocación de objetos

Con la herramienta de pinza, crear un programa para coger el objeto en la base y soltarlo en la plantilla por orden cada vez que se detecte un objeto. Tras soltar el objeto, volver a origen y esperar a que haya otro objeto. Terminar el programa cuando se hayan colocado 5 objetos.

Las operaciones de cogida y suelta del objeto han de realizarse con aproximaciones en Z de base y con comprobaciones de los finales de carrera.



Siguiente bloque:

Cabezal de
succión

