

EPSON

Portal AX

Manual del usuario

Versión traducida

©Seiko Epson Corporation 2026

Rev. 2
ESM265S8841M

Índice

Marcas.....	8
Condiciones de uso.....	8
1. Introducción.....	9
1.1 Programación y control.....	10
1.2 Interfaz gráfica de usuario (GUI).....	10
1.3 Módulos de software.....	10
1.3.1 Módulo de conexión TCP.....	10
1.3.2 Módulo de conexión REST.....	11
1.3.3 Módulo de conexión ROS.....	11
1.3.4 Módulo de conexión Modbus.....	11
1.3.5 Interfaz del panel.....	11
1.3.6 Módulo de cámara Cognex.....	11
2. Interfaz de administración (GUI).....	12
2.1 Introducción.....	13
2.1.1 Idioma de la interfaz.....	14
2.1.2 Panel de control.....	14
2.2 Gestión de usuarios.....	14
2.2.1 Niveles de autorización.....	14
2.2.2 Página de inicio de sesión.....	14
2.2.3 Gestionar su propia cuenta.....	15
2.2.4 Gestionar otras cuentas de usuario.....	16
2.3 Configuración del sistema.....	17
2.3.1 Configuración de red.....	17
2.3.2 Ajustes de hora.....	18
2.3.3 Certificado SSL.....	18
2.4 Actualización de software.....	20
2.5 Ayuda y asistencia.....	22
2.6 Autenticación segura.....	23
2.6.1 Exportar certificado SSL.....	23
2.6.2 Inicio de sesión del usuario.....	23
2.6.3 Registro de la conexión.....	24
3. Exportación a USB y actualización de la copia de seguridad.....	25
3.1 Resumen.....	26
3.2 Exportación USB.....	26
3.3 Ejecutar la actualización de recuperación.....	26
4. Interfaz de escritorio (GUI).....	27
4.1 Introducción.....	28
4.2 Ventana principal.....	29
4.2.1 Resumen de menús.....	30
4.2.2 Panel de control.....	30
4.2.3 Control activo (Manejo exclusivo desde una estación de control).....	32
4.2.3.1 Control activo.....	32

4.2.3.2 Observador pasivo.....	33
4.2.4 Estado del programa.....	33
4.2.5 Modos de funcionamiento.....	35
4.2.5.1 Modo automático.....	35
4.2.5.2 Modo manual.....	35
4.2.6 Control de precisión.....	36
4.2.7 Control de movimiento.....	37
4.2.7.1 Límites de movimiento.....	37
4.2.7.2 Parámetros globales.....	38
4.2.7.3 Sistemas de coordenadas globales.....	39
4.3 Bloques y base de datos.....	40
4.4 Menú de proyectos.....	41
4.5 Menú de configuración.....	42
4.5.1 Base de datos (descarga y carga).....	42
4.5.2 Configuración de E/S.....	43
4.5.3 Configuración de la herramienta.....	45
4.5.4 Editor de herramientas.....	45
4.5.4.1 Modo de salida única.....	47
4.5.4.2 Modo de salida doble.....	47
4.5.4.3 Modo de activación separada.....	47
4.5.5 Puesta en marcha del robot.....	48
4.6 Menú de aprendizaje.....	49
4.6.1 Editor de poses.....	51
4.6.2 Editor de trayectorias.....	53
4.6.3 Editor de cuadrículas.....	55
4.7 Menú de aplicaciones.....	58
4.7.1 Editor de aplicaciones.....	59
4.7.2 Código del programa.....	59
4.7.3 Configuración de la aplicación.....	61
4.7.4 Variables de la aplicación.....	63
4.7.5 Variables compartidas y globales.....	65
4.7.6 Editor de carpetas.....	66
4.7.7 Programación por bloques.....	67
4.7.7.1 Uso de variables de aplicación y expresiones.....	68
4.7.7.2 Bloque de movimiento.....	68
4.7.7.3 Bloque de trayectoria.....	69
4.7.7.4 Bloque de recogida.....	69
4.7.7.5 Bloque «Colocar».....	70
4.7.7.6 Bloque «Contenedor».....	70
4.7.7.7 Bloque «Siguiendo pose de rejilla».....	70
4.7.7.8 Bloque «Restablecer pose de rejilla».....	70
4.7.7.9 Bloque «Escribir variable».....	70
4.7.7.10 Bloque «Establecer salida».....	71
4.7.7.11 Bloque Esperar.....	71

4.7.7.12 Bloque «Bucle».....	71
4.7.7.13 Bloque «Salir si».....	71
4.7.7.14 Bloque de condición.....	71
4.7.7.15 Bloque de comentario.....	71
4.7.7.16 Bloque «Finalizar».....	71
4.7.7.17 Bloque de diálogo.....	72
4.7.7.18 Bloque de salida.....	72
4.7.7.19 Bloque de registro.....	72
4.7.7.20 Bloque de notificación.....	72
4.7.7.21 Bloque de advertencia.....	72
4.7.7.22 Bloque de error.....	72
4.7.7.23 Bloque de código Python.....	72
4.8 Menú E/S.....	72
4.9 Menú de visualización.....	74
4.10 Control manual.....	75
4.11 Monitor del sistema.....	76
4.11.1 Estado del robot.....	76
4.11.2 Salida de texto.....	76
4.11.3 Registro de aplicaciones.....	76
4.11.4 Historial de mensajes.....	76
4.12 Menú de ayuda.....	77
4.12.1 Información del robot.....	77
4.12.2 Atajos de teclado.....	77
4.12.3 Documentación del código.....	77
4.12.4 Licencia de código abierto.....	77
4.12.5 Información de contacto.....	77
4.13 Menú de configuración.....	78
4.13.1 Ajustes de la interfaz gráfica de usuario.....	79
4.13.2 Personalizar botones del panel de control.....	79
4.13.3 Ajustes de inicio del programa.....	80
4.13.4 Comportamiento de movimiento.....	80
4.13.4.1 Comportamiento ante interrupciones.....	80
4.13.4.2 Cuadro de diálogo de interrupción.....	81
4.14 Menú de configuración de seguridad.....	84
4.14.1 Interfaz de usuario.....	84
4.14.2 Ajustes de seguridad.....	86
4.14.2.1 Posición limitada de seguridad (SLP).....	86
4.14.2.2 Ángulos de articulación limitados de forma segura.....	87
4.14.2.3 Velocidad limitada de forma segura (SLS).....	88
4.14.2.4 Ajustes de seguridad para potencia y fuerza.....	89
4.14.2.5 E/S de seguridad y restricciones de usuario.....	90
4.14.2.6 Datos de verificación.....	91
4.14.2.7 Funciones de salida de seguridad.....	91
4.14.2.8 Conexión de dispositivos opcionales.....	91

5. Interfaz del panel (GUI).....	92
5.1 Introducción.....	93
5.2 Configuración de aplicaciones para la interfaz del panel.....	94
5.3 Panel de control y ajustes.....	96
5.4 Control de aplicaciones.....	97
6. Definiciones cinemáticas.....	98
6.1 Coordenadas.....	99
6.1.1 Ángulos náuticos.....	99
6.1.2 Sistemas de referencia.....	100
6.2 Configuración.....	102
6.3 Postura.....	102
6.4 Punto.....	102
6.5 Trayectoria.....	102
6.6 Segmento.....	103
6.7 Rejillas.....	103
7. Módulos de conexión.....	105
7.1 Conexiones, autenticación y control activo.....	106
7.1.1 Canales de conexión.....	106
7.1.2 Autenticación.....	106
7.1.3 Control activo.....	107
7.2 Módulo de conexión TCP.....	107
7.2.1 Establecer conexión.....	107
7.2.2 Envío de comandos.....	109
7.2.3 Configuración.....	110
7.2.3.1 Estándar de codificación.....	110
7.2.3.2 Carácter de fin.....	110
7.2.3.3 Enviar latido.....	110
7.2.4 Protocolo de mensajes: recepción de comandos.....	111
7.2.4.1 Información de estado.....	111
7.2.4.2 Valor de retorno de la acción.....	112
7.2.4.3 Errores.....	112
7.2.4.4 Historial de mensajes.....	112
7.2.4.5 Definición de un grupo de estados.....	112
7.2.4.6 Datos de un grupo de estado.....	113
7.2.4.7 Abrir un cuadro de diálogo.....	114
7.2.4.8 Enviar desde la aplicación.....	114
7.2.5 Registro de mensajes: envío de comandos.....	115
7.2.5.1 Inicio de sesión con token.....	115
7.2.5.2 Solicitar el control activo.....	115
7.2.5.3 Liberar el control activo.....	116
7.2.5.4 Consultar datos.....	116
7.2.5.5 Puesta en marcha.....	117
7.2.5.6 Poner fuera de servicio.....	117
7.2.5.7 Pausar.....	117

7.2.5.8 Reanudar.....	118
7.2.5.9 Activación del control manual (HGC).....	118
7.2.5.10 Desactivación del control manual (HGC).....	119
7.2.5.11 Detener.....	119
7.2.5.12 Cambiar de proyecto.....	120
7.2.5.13 Ejecutar una aplicación.....	120
7.2.5.14 Probar la aplicación.....	120
7.2.5.15 Recibir un mensaje.....	121
7.2.5.16 Desplazarse a una pose.....	121
7.2.5.17 Reproducir ruta.....	123
7.2.5.18 Cambiar herramienta.....	123
7.2.5.19 Escribir salidas digitales y analógicas.....	124
7.2.5.20 Responder a un cuadro de diálogo.....	125
7.2.5.21 Establecer variable de estado.....	125
7.2.5.22 Establecer una variable global.....	126
7.2.5.23 Leer variable global.....	126
7.2.5.24 Establecer una variable compartida.....	127
7.2.5.25 Leer variable compartida.....	127
7.3 Módulo de conexión REST.....	128
7.3.1 Establecer conexión.....	128
7.3.2 Envío de comandos.....	129
7.3.3 Configuración.....	129
7.4 Módulo de conexión ROS.....	130
7.4.1 Instalación.....	130
7.4.2 Configuración.....	131
7.4.3 Detección de red con Cyclone DDS.....	131
7.4.3.1 Interfaz de red.....	131
7.4.3.2 Métodos de detección.....	132
7.4.4 Establecer la conexión.....	134
7.4.5 Acceso a los datos del robot a través de ROS.....	135
7.4.6 Funcionalidades de ROS 2 en scripts.....	135
7.4.7 Llamar a una acción.....	135
7.4.8 Configuración de autenticación.....	136
7.5 Módulo de conexión Modbus.....	136
7.5.1 Resumen.....	136
7.5.2 Configuración.....	137
7.5.3 Registros de escritura (descripción sencilla).....	138
7.5.3.1 Registro de comandos.....	138
7.5.3.2 Registro de propiedades.....	139
7.5.3.3 Registros de escritura definidos por el usuario.....	139
7.5.4 Registros de escritura (descripción detallada).....	140
7.5.4.1 No realizar ninguna acción.....	140
7.5.4.2 Puesta en marcha.....	140
7.5.4.3 Poner fuera de servicio.....	140

7.5.4.4 Parada.....	140
7.5.4.5 Pausa.....	140
7.5.4.6 Continuar.....	141
7.5.4.7 Calibración.....	141
7.5.4.8 Activar el control manual (HGC).....	141
7.5.4.9 Desactivar el modo de control manual (HGC).....	141
7.5.4.10 Cambiar de proyecto.....	141
7.5.4.11 Iniciar la aplicación.....	142
7.5.4.12 Desplazarse a la pose.....	142
7.5.4.13 Reproducir ruta.....	142
7.5.4.14 Colocar herramienta.....	142
7.5.4.15 Levantar la herramienta.....	142
7.5.4.16 Escritura en salidas digitales.....	143
7.5.4.17 Configuración de eventos de aplicación.....	143
7.5.4.18 Confirmar errores.....	143
7.5.4.19 Lectura de la posición.....	143
7.5.4.20 Solicitar control activo.....	143
7.5.4.21 Liberar el control activo.....	144
7.5.4.22 Cambiar al modo manual.....	144
7.5.4.23 Cambiar al modo automático.....	144
7.5.4.24 Restablecimiento de seguridad.....	144
7.5.4.25 Iniciar la interrupción del movimiento.....	145
7.5.5 Registros de lectura.....	146
7.5.5.1 Registro de estado.....	146
7.5.5.2 Registros de lectura definidos por el usuario.....	149
7.5.5.3 Registros de posición (articulaciones 1 a 6).....	149
7.5.5.4 Registro de coordenadas (x, y, z, roll, pitch, yaw).....	149
7.5.6 Asignación de los registros de error.....	150
7.5.6.1 Ejemplo.....	151
7.5.7 Ejemplos de código Python.....	152
7.5.7.1 Ejemplo 1.....	152
7.5.7.2 Ejemplo 2.....	152
7.5.7.3 Ejemplo 3.....	153
7.5.7.4 Ejemplo 4.....	153
8. Módulo de cámara Cognex.....	154
8.1 Descripción general.....	155
8.2 Configuración de una cámara.....	155
8.3 Uso de una cámara.....	155

Marcas

Cognex® e In-Sight Explorer® son marcas comerciales o marcas registradas de Cognex Corporation en Estados Unidos y otros países.

Google Chrome™ y Chrome™ son marcas comerciales de Google LLC.

Linux® es una marca registrada de Linus Torvalds en EE. UU. y otros países.

Apple®, macOS® y Safari® son marcas registradas de Apple Inc. en Estados Unidos y en otros países y regiones.

Microsoft®, Windows®, Microsoft Edge® y Microsoft Visual C++® son marcas (o marcas registradas) de Microsoft Corporation y/o sus filiales en Estados Unidos y otros países.

Mozilla y Firefox® son marcas de Mozilla Foundation en EE. UU. y otros países.

Nota general: Los nombres de productos utilizados aquí se emplean únicamente con fines identificativos y pueden ser marcas de sus respectivos propietarios. Epson no reclama ningún derecho sobre estas marcas.

Todas las capturas de pantalla se utilizan exclusivamente con fines ilustrativos y formativos. No se pretende ni se da a entender ninguna relación con el titular de la marca ni su aprobación.

Condiciones de uso

Ninguna parte de este manual de instrucciones puede ser reproducida o reimpressa en forma alguna sin autorización expresa por escrito.

La información contenida en este documento está sujeta a cambios sin previo aviso.

Póngase en contacto con nosotros si detecta algún error en este documento o si tiene alguna pregunta sobre la información que contiene.

1. Introducción

1.1 Programación y control

El marco de software AX Portal ofrece al usuario de un robot de la serie Epson AX una forma muy intuitiva de comunicarse con dicho robot. Con AX Portal, el usuario dispone de acceso completo para manipular y controlar el robot, así como para escribir aplicaciones de programa con el lenguaje de programación [Python 3.13](#). La funcionalidad básica para la programación con Python se encuentra disponible en línea. No obstante, en la documentación de código integrada también se proporcionan algunos ejemplos de código muy básicos.

Además de una amplia selección de funciones básicas de Python, las aplicaciones de AX Portal admiten una extensa colección de funciones diseñadas específicamente para el control, el mantenimiento y la comunicación con los robots AX. Encontrará la lista completa de funciones de script compatibles con AX Portal en la documentación de código integrada.

1.2 Interfaz gráfica de usuario (GUI)

La GUI se ha diseñado como una aplicación web que puede ejecutarse en los navegadores web habituales de prácticamente cualquier dispositivo. Los siguientes navegadores son compatibles oficialmente:

- Google Chrome (versión ≥ 147)
- Mozilla Firefox (versión ≥ 150)
- Apple Safari (versión ≥ 26)
- Microsoft Edge (versión ≥ 147)

Por lo tanto, no es necesario descargar un módulo de software específico para su controlador, siempre y cuando el robot y el dispositivo se encuentren en la misma red o compartan un canal de comunicación directo. Además de la interfaz de escritorio estándar de AX Portal, optimizada principalmente para la programación en ordenadores de sobremesa y portátiles, AX Portal ofrece una interfaz de panel que contiene únicamente las funciones necesarias para manejar un robot ya preprogramado. Encontrará más información sobre las interfaces gráficas de usuario y todas las funciones compatibles más adelante en este documento.

1.3 Módulos de software

Además de la funcionalidad básica de AX Portal, que se describe detalladamente en este documento, los robots AX disponen de varios módulos de software integrados que ofrecen opciones de control adicionales. Tenga en cuenta que todos los módulos de conexión mencionados a continuación permiten conectarse de forma remota al robot AX, lo que ofrece posibilidades para controlar y supervisar el robot.

1.3.1 Módulo de conexión TCP

El módulo de conexión TCP puede utilizarse como una opción alternativa para comunicarse con el robot. Por lo general, el usuario controla el robot a través de la interfaz gráfica de usuario (GUI) en el navegador web. Sin embargo, si se desea controlar el robot a través de una interfaz definida por el usuario o de otra máquina, este acepta comandos TCP directamente. Con el módulo de conexión TCP,

todo lo que está disponible a través de la comunicación por navegador también se puede acceder a través de TCP.

1.3.2 Módulo de conexión REST

El módulo de conexión REST puede utilizarse como alternativa para comunicarse con el robot. Por lo general, el usuario controla el robot a través de la interfaz gráfica de usuario (GUI) en el navegador web. Sin embargo, si se desea controlar el robot a través de una interfaz definida por el usuario o de otra máquina, este acepta comandos REST directamente. Con el módulo de conexión REST, todo lo que está disponible a través de la comunicación por navegador también puede accederse a través de REST.

1.3.3 Módulo de conexión ROS

El módulo de conexión ROS ofrece funcionalidades para integrar el robot en un entorno ROS2. El robot permite que otros clientes de la red ROS2 envíen comandos para controlar y supervisar el robot.

1.3.4 Módulo de conexión Modbus

El módulo de conexión Modbus puede utilizarse como alternativa para comunicarse con el robot. Utiliza el protocolo Modbus-TCP y permite al usuario leer y escribir registros Modbus estándar. Utilice este módulo para controlar el robot directamente a través de un PLC. Las funciones más importantes, como la puesta en marcha del robot, la ejecución de aplicaciones y la activación de salidas, se pueden realizar a través del módulo Modbus.

1.3.5 Interfaz del panel

La interfaz del panel ofrece una interfaz gráfica de usuario adicional con funciones limitadas, pensada más para los operadores de robots que para los programadores. Esta interfaz sirve principalmente para iniciar y supervisar aplicaciones, y el programador puede adaptar su contenido a su caso de uso. La interfaz del panel está optimizada para dispositivos con pantalla táctil y su diseño se asemeja a la conocida estructura tipo aplicación en dispositivos móviles modernos, lo que la hace intuitiva incluso para usuarios sin conocimientos técnicos.

1.3.6 Módulo de cámara Cognex

El módulo de cámara Cognex proporciona una interfaz de comunicación entre una cámara Cognex y AX Portal. Las cámaras suelen estar integradas en un efector final o montadas de forma fija para observar la escena en la que trabaja el robot. Este módulo es ideal para la inspección visual personalizable y es compatible con todas las funciones de las cámaras Cognex.

2. Interfaz de administración (GUI)

2.1 Introducción

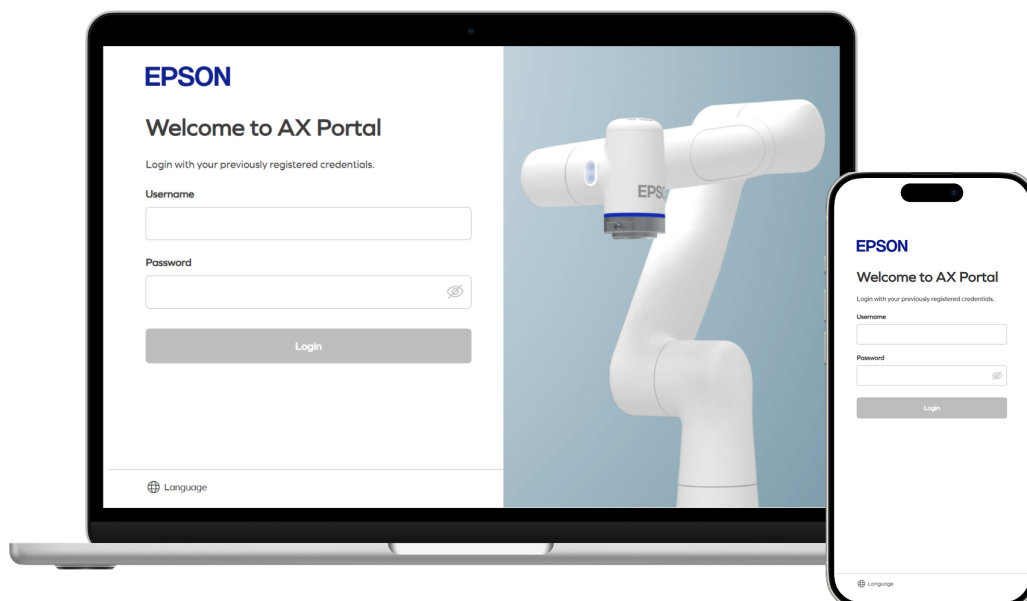


Figura 2.1.a: Ejemplos de capturas de pantalla de la interfaz de administración

La interfaz de administración de AX Portal es una interfaz gráfica de usuario (GUI) que permite a los usuarios gestionar cuentas de usuario y modificar algunos ajustes básicos vinculados al sistema operativo del robot (como la configuración de red, la configuración de la hora, etc.). Por lo general, estos ajustes se configuran una sola vez durante la instalación del robot, incluso antes de que un programador comience a escribir aplicaciones de control del robot. Se puede acceder a esta interfaz a través de un navegador web (por ejemplo, Google Chrome) introduciendo la dirección IP del robot en la barra de direcciones del navegador, lo que le redirigirá a <https://<DIRECCIÓN IP>/admin> (consulte la «Guía del controlador del robot RC-A101» para obtener la dirección IP correcta).

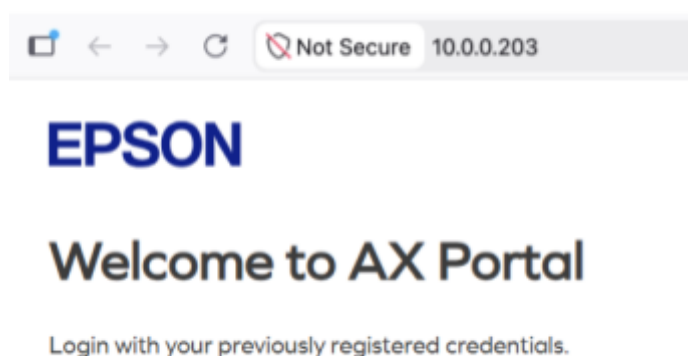


Figura 2.1.b: Conexión a la interfaz de administración a través de la URL

En este capítulo se explican detalladamente todos los menús y elementos de la interfaz de administración.

2.1.1 Idioma de la interfaz

Para cambiar el idioma de la interfaz, haga clic en el menú desplegable de idiomas situado en la esquina inferior izquierda de la interfaz y seleccione su idioma preferido. Su navegador le facilitará el trabajo al guardar su configuración de idioma en todas las sesiones e interfaces.

2.1.2 Panel de control

Tras iniciar sesión, el panel de control de la interfaz de administración le permite navegar por los distintos submenús, acceder a otras interfaces de AX Portal y cerrar la sesión. Tenga en cuenta que el panel de control solo muestra las opciones a las que tiene acceso actualmente, en función de su nivel de usuario (administrador, responsable de seguridad, programador, operador). Si no puede navegar a otras interfaces, AX Portal no está funcionando correctamente en este momento, lo cual ocurre, por ejemplo, mientras se lleva a cabo una actualización de software.

2.2 Gestión de usuarios

2.2.1 Niveles de autorización

Para evitar que el robot sea configurado, programado o manejado por usuarios no autorizados, se distinguen diferentes niveles de autorización. La autorización para cada usuario la establecen exclusivamente los administradores. Están disponibles los siguientes niveles de autorización, de mayor a menor:

- Un **administrador** tiene acceso completo a todas las interfaces y funciones de control del robot, así como a la gestión de otros usuarios y a los ajustes más importantes del sistema operativo.
- Un **programador** tiene acceso a todas las interfaces y funciones de control del robot y, por lo tanto, puede preparar y programar aplicaciones que posteriormente ejecutarán los operadores. Dentro de la interfaz de administración, el programador tiene acceso únicamente a los datos de su propia cuenta y a la navegación entre las interfaces.
- Un **responsable de seguridad** es un programador (véase más arriba) que está autorizado a modificar la configuración de seguridad del robot. El resto de usuarios solo pueden consultar la configuración de seguridad.
- Un **operador** solo tiene acceso a las interfaces básicas de operador que permiten el manejo del robot, pero no puede modificar la configuración ni los programas. Dentro de la interfaz de administración, el operador tiene acceso únicamente a los datos de su propia cuenta.

2.2.2 Página de inicio de sesión

La página de inicio de sesión contiene dos formularios diferentes, dependiendo de si ya se ha creado una cuenta de administrador.

- Si ya se ha creado un administrador, aparecerá la página de inicio de sesión habitual, en la que deberá introducir su nombre de usuario y contraseña para iniciar sesión. Las contraseñas se ocultan al introducirlas, a menos que haga clic en el botón para mostrar la contraseña en el campo de contraseña.
- Si aún no se ha creado ningún administrador, se le pedirá directamente que lo haga, y solo podrá continuar después de haber creado una cuenta de administrador. Encontrará más

detalles en la «Guía rápida / Manual de instalación del robot colaborativo de 6 ejes AX6». Aquí solo puede especificar los datos estrictamente necesarios; si desea ver y definir todos los ajustes de su cuenta, acceda al menú «Manage My Account».

Cada vez que cierre la sesión, se le redirigirá a la página de inicio de sesión.

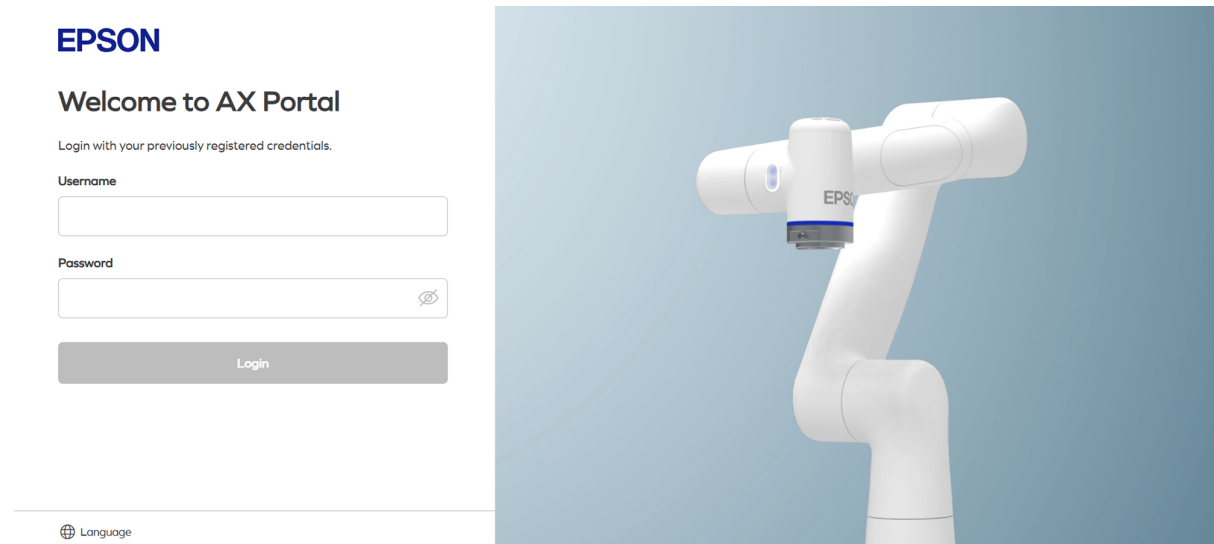


Figura 2.2.2: Página de inicio de sesión

2.2.3 Gestionar su propia cuenta

En el menú «Manage My Account» puede modificar su propio perfil, concretamente su nombre, su contraseña, su foto de perfil y la pantalla de inicio:

- El «Nombre completo» es el nombre que se utiliza para dirigirse al usuario en la interfaz y en los mensajes. Se trata de un campo obligatorio destinado a su nombre y apellidos; no obstante, también puede establecer sus propias convenciones de nomenclatura.
- El campo «Nombre de usuario» no se puede modificar una vez creado el usuario; el administrador establece el nombre de usuario al crear la cuenta.
- El campo «Interfaz de inicio» determina a qué interfaz desea que se le redirija tras iniciar sesión. Para los distintos niveles de autorización hay predefinidas diferentes interfaces de inicio predeterminadas, que puede modificar aquí.
- Mediante el botón «Cambiar contraseña» puede modificar la contraseña de su cuenta. Se le pedirá que vuelva a introducir su contraseña actual y que introduzca su nueva contraseña dos veces, con el fin de minimizar los errores de introducción accidentales al definir la contraseña. Su contraseña también puede ser modificada por cualquier administrador en caso de que la olvide. Las contraseñas nunca se almacenan en texto claro; por lo tanto, ni usted ni ningún otro usuario podrá leer su contraseña.
- La «Imagen de perfil» se puede cambiar de forma opcional. Se mostrará un icono de usuario predeterminado si no define una imagen de perfil.

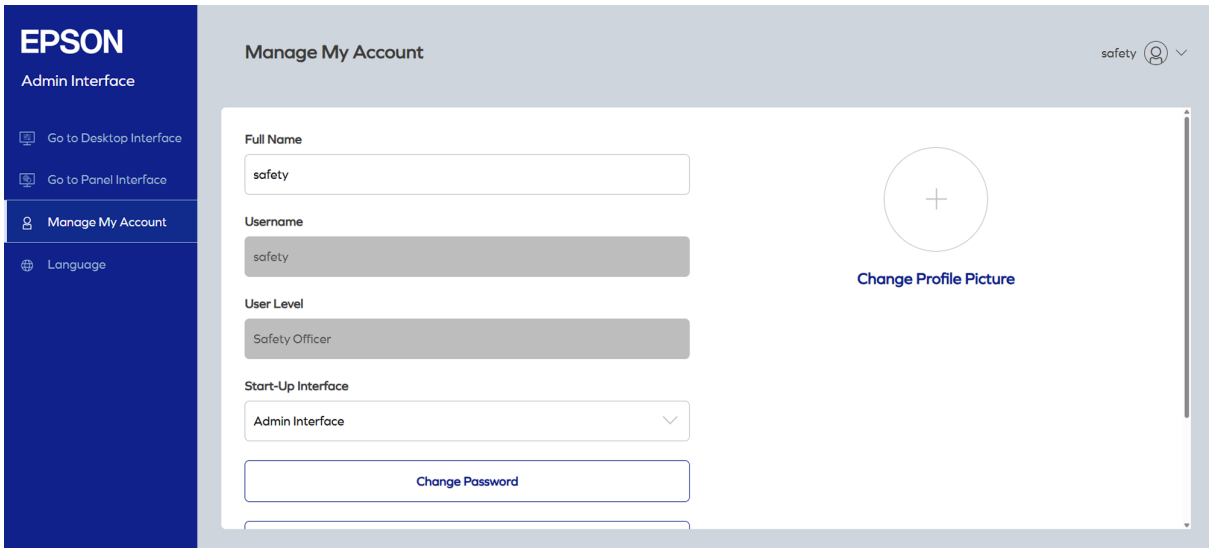


Figura 2.2.3: Página «Gestionar su cuenta»

2.2.4 Gestionar otras cuentas de usuario

En el menú «Otras cuentas de usuario» verá un resumen de todos los usuarios disponibles actualmente. Aquí dispone de las siguientes opciones:

- La opción «Añadir un nuevo usuario» le permite crear un usuario con un nivel de autorización y una contraseña inicial predefinidos. Cada usuario debe cambiar la contraseña inicial tras su primer inicio de sesión.
- La opción «Modificar un usuario» le permite cambiar la mayoría de las propiedades de dicho usuario, a excepción del nombre de usuario, que no se puede modificar.
- Al eliminar un usuario, este se borra de forma permanente de la base de datos. A partir de ese momento, dicho usuario ya no podrá activar ninguna función de control del robot.

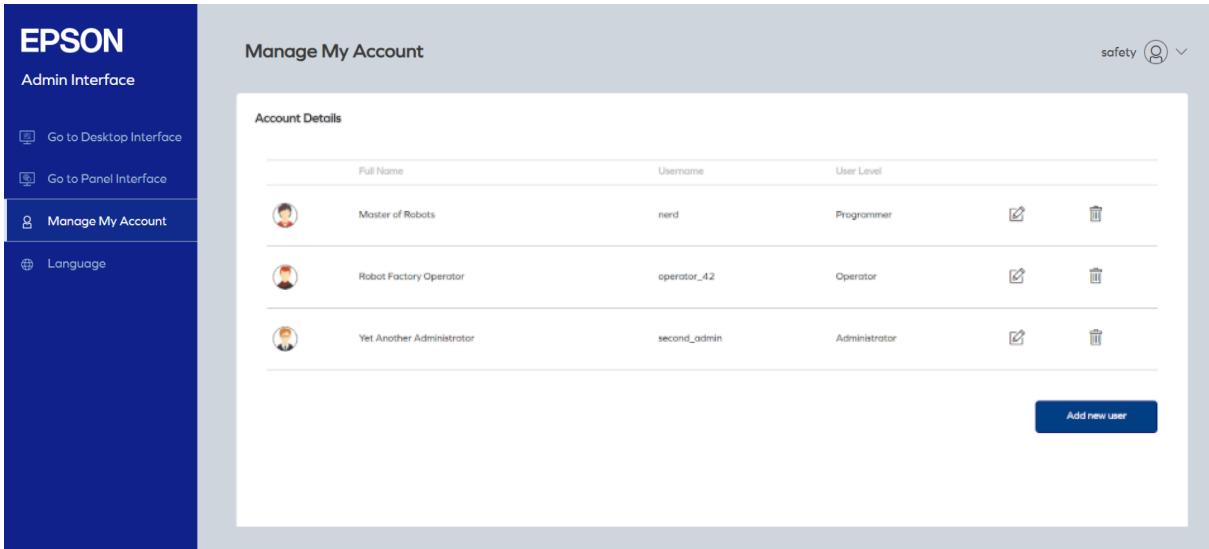


Figura 2.2.4: Página «Gestionar otras cuentas de usuario»

2.3 Configuración del sistema

2.3.1 Configuración de red

El menú «Configuración de red» contiene todos los ajustes relevantes para definir la conexión Ethernet del robot. Tenga en cuenta que perderá la conexión tan pronto como modifique la configuración de la conexión actualmente activa (por ejemplo, si está conectado a través de Ethernet y cambia la dirección IP de Ethernet). Si introduce accidentalmente una configuración no válida, es posible que ya no pueda volver a conectarse al robot. Por lo tanto, se recomienda altamente asegurarse de que funcionan dos conexiones diferentes antes de modificar la configuración de la única conexión que funciona.

La configuración de Ethernet muestra y le permite decidir si el robot utiliza una dirección IP dinámica o manual. Si el robot está configurado para obtener una dirección IP de forma dinámica, aquí también verá la dirección IP seleccionada. Si el robot está configurado manualmente, deberá introducir una dirección IP y la máscara de red (campo obligatorio); además, puede definir la puerta de enlace y hasta dos servidores DNS (opcional).

The screenshot shows the EPSON Admin Interface with a sidebar on the left containing the following menu items: Go to Desktop Interface, Go to Panel Interface, Manage My Account, Manage Other Accounts, Network Settings (highlighted), Time Settings, SSL Certificate, Software Update, Help & Support, and Language. The main content area is titled 'Network Settings' and features a 'User' dropdown in the top right corner. Under the 'Ethernet IPv4' section, there are two radio buttons: 'Dynamic IP Address' (selected) and 'Manual IP Address'. Below these are input fields for 'IP Address' (192.168.90.200), 'Netmask' (255.255.254.0), 'Gateway' (192.168.90.1), 'DNS Server 1', and 'DNS Server 2'. A 'Network Information' section contains a note about using the second Ethernet port for maintenance. At the bottom are 'Cancel' and 'Save' buttons.

Ethernet IPv4	
IP Address	192.168.90.200
Dynamic IP Address	☒
Manual IP Address	☐
IP Address	192.168.90.200
Netmask	255.255.254.0
Gateway	192.168.90.1
DNS Server 1	
DNS Server 2	

Network Information

You may use the second Ethernet port to access AX Portal for network maintenance reasons. Its static IP address is 10.0.0.203 and cannot be changed by the administrator. You may connect to "<Robot-Serial>.local" instead of entering the IP address to connect to AX Portal via your browser.

Cancel Save

Figura 2.3.1: Página «Configuración de red»

2.3.2 Ajustes de hora

El menú «Configuración de hora» muestra la fecha actual, la hora actual y la zona horaria del sistema operativo del robot. Asegúrese de introducir la hora y la fecha correctas. Al establecer la hora correcta, el sistema podrá utilizarse para el análisis de registros y la resolución de errores, y las dependencias temporales funcionarán correctamente. Si cambia la hora manualmente, la hora exacta se aplicará inmediatamente después de pulsar el botón «Guardar». Si desea cambiar la zona horaria, puede filtrar la lista de zonas horarias disponibles en el sistema por continente (p. ej., «Europe»), código de país (p. ej., «GB»), ciudad importante (p. ej., «London») o una combinación de estos (p. ej., «eur lon» → Europe/London).

Nota: Si el robot dispone de acceso a la red, puede activar la sincronización NTP (Network Time Protocol).

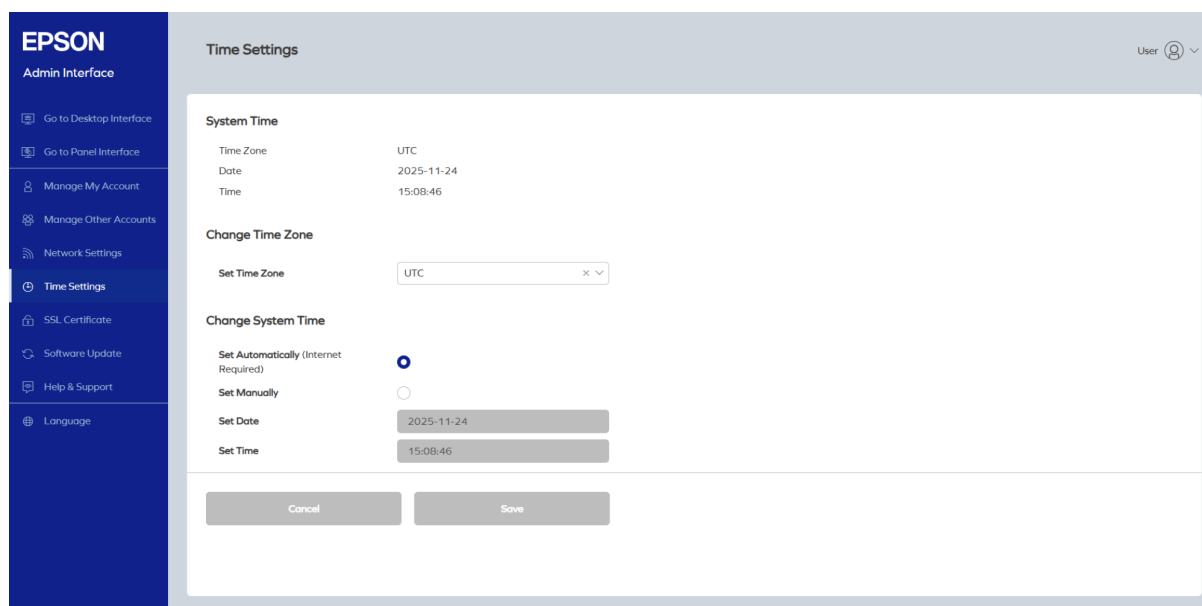


Figura 2.3.2: Página «Configuración de la hora»

2.3.3 Certificado SSL

El menú «Certificado SSL» muestra algunos detalles sobre el certificado SSL cargado actualmente y le permite sustituir este certificado por su propio certificado privado o empresarial. Un certificado SSL (Secure Sockets Layer) es un certificado digital que autentifica la identidad de un sitio web y permite una conexión cifrada. Este certificado garantiza que las transacciones en línea y la información de los clientes permanezcan privadas y seguras en su navegador web. De forma predeterminada, se instala un certificado autofirmado, que debe sustituirse por un certificado adecuado al entorno del usuario y firmado por una entidad de confianza.

Los navegadores marcan un certificado como no seguro si

- el nombre de dominio (o la dirección IP) registrado en el certificado no coincide con el dominio (o la dirección IP) que se utiliza para acceder al robot,
- el certificado ha caducado, o
- el certificado es autofirmado.

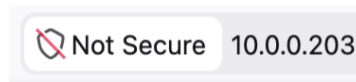


Figura 2.3.3.a: Ejemplo de un navegador que señala un certificado SSL inseguro o ausente

Para sustituir el certificado SSL, debe cargar tanto la clave privada como la pública.

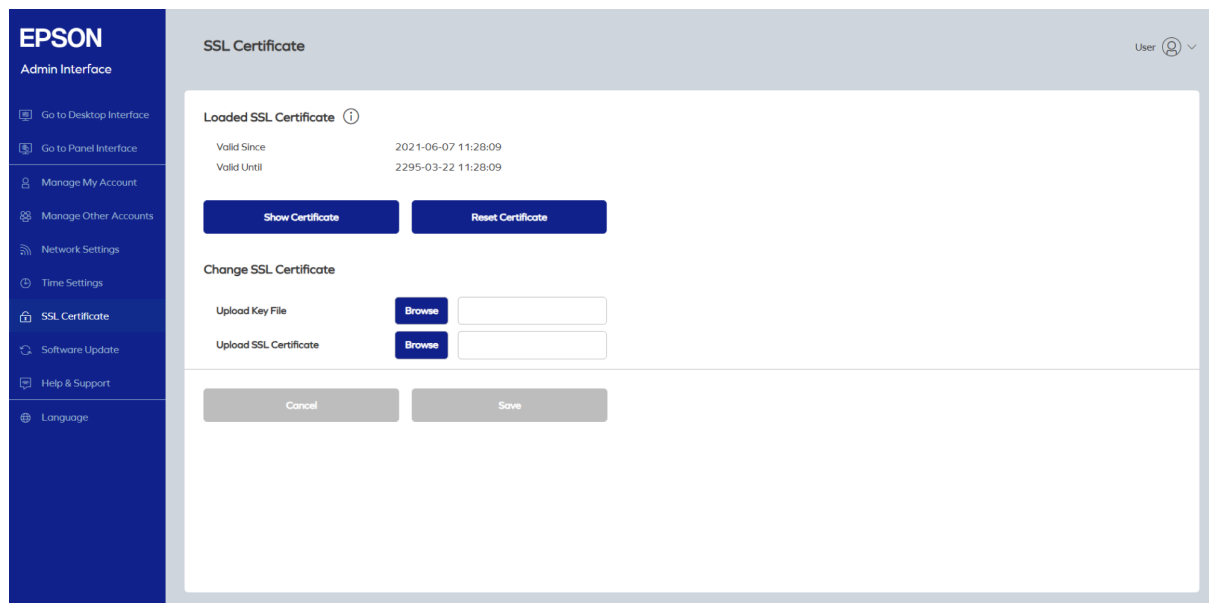


Figura 2.3.3.b: Página «Certificado SSL»

2.4 Actualización de software

En el menú «Actualización de software» podrá ver la versión actual de AX Portal. Aquí puede cargar e instalar nuevas versiones de software. Es posible que sea necesaria una actualización de software para solucionar un problema concreto o para obtener las últimas funciones y correcciones de errores.

Antes de actualizar la versión del software, lea siempre primero las notas de la versión, ya que contienen información importante sobre los posibles riesgos de actualizar a determinadas versiones. Las mejoras a veces se producen a expensas de funciones antiguas que se consideran obsoletas. Le recomendamos que realice una copia de seguridad de su base de datos, que incluya todas las aplicaciones y configuraciones de los robots (a través de la interfaz de escritorio), antes de iniciar la actualización.

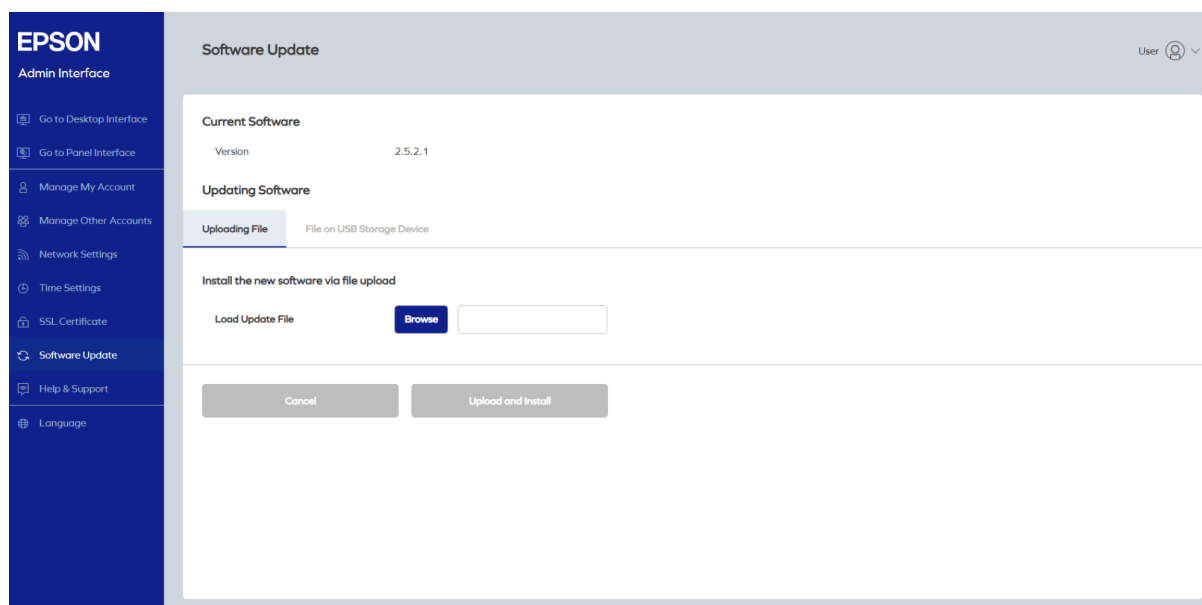


Figura 2.4.a: Página «Actualización de software»

Para instalar una actualización de software, primero debe obtener el archivo RBIN que contiene la actualización para la versión de AX Portal que desea instalar. Existen tres formas de instalar esta actualización:

- 1) Seleccione la primera pestaña «Cargar archivo». Aquí puede cargar la actualización desde su dispositivo. Explore el sistema de archivos y seleccione el archivo RBIN de actualización; a continuación, continúe con la instalación de la actualización.
- 2) Seleccione la segunda pestaña «Archivo desde dispositivo USB». Aquí puede guardar el archivo RBIN en una memoria USB y conectarla al controlador del robot. Seleccione el archivo RBIN de actualización del directorio de la memoria USB y, a continuación, continúe con la instalación.
Nota: Actualmente se admiten los sistemas de archivos exFAT y FAT32 para la memoria USB.
- 3) Si las opciones 1) y 2) no son aplicables (por ejemplo, si la interfaz de administración de su robot no funciona correctamente), siempre tiene la posibilidad de ejecutar la actualización de recuperación en su lugar.

Mientras se realiza una actualización, podrá ver qué paso se está ejecutando en ese momento, qué pasos se han completado con éxito y, en caso de error, en qué paso se ha producido el error, junto con el mensaje de error. Una actualización puede durar **hasta 15 minutos**. Por lo tanto, le rogamos que tenga paciencia y no interrumpa la actualización, por ejemplo, desconectando la alimentación eléctrica del robot. En el peor de los casos, esto podría provocar que posteriormente ya no pueda acceder al robot.

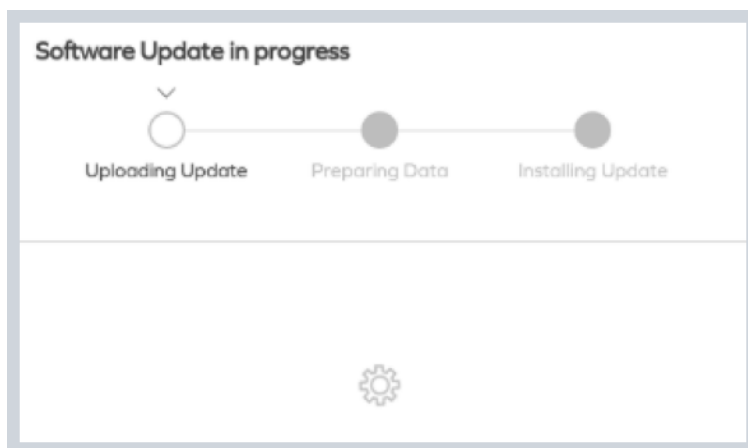


Figura 2.4.b: Ventana «Actualización de software en curso»

Una vez completada con éxito una actualización, esta debe confirmarse. Hasta que se confirme la actualización, se mostrará un mensaje en rojo. Reiniciar el controlador restablecerá la actualización si esta no se ha confirmado. Una vez confirmada, ya no será posible volver a una versión anterior.

2.5 Ayuda y asistencia

El menú «Ayuda y asistencia» muestra la información de contacto que puede utilizar para ponerse en contacto con Epson en caso de que necesite ayuda. Le rogamos que consulte primero la documentación antes de solicitar asistencia por correo electrónico o por teléfono.

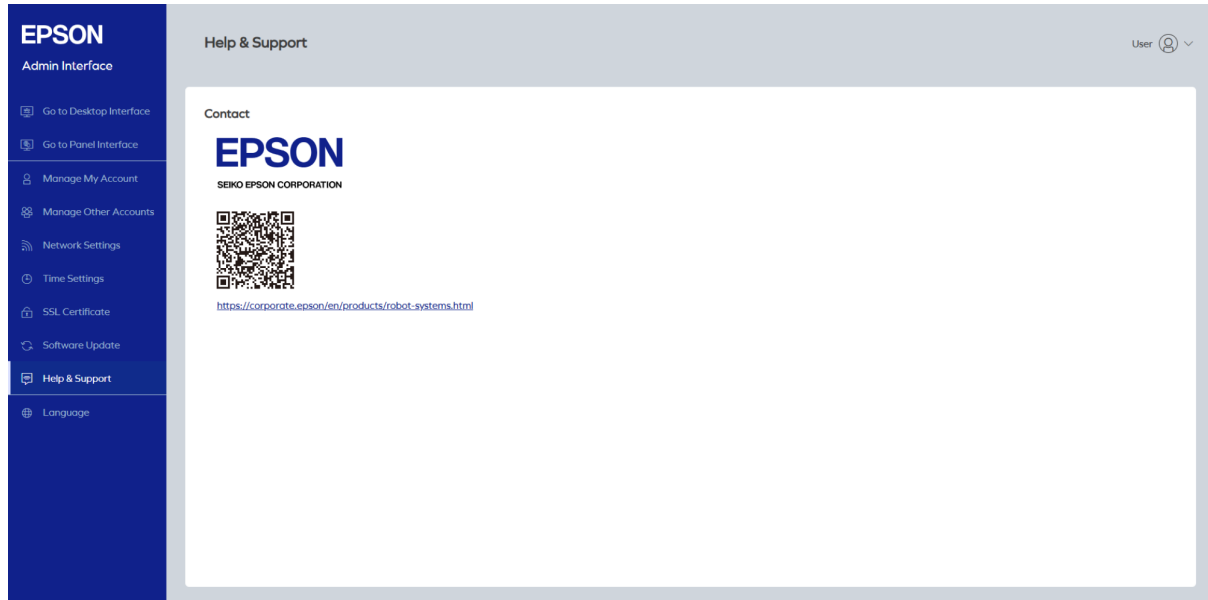


Figura 2.5: Página «Ayuda y asistencia»

2.6 Autenticación segura

Si utiliza módulos de conexión para acceder a AX Portal y controlar el robot (sin GUI), deberá autenticarse mediante un procedimiento seguro de dos pasos antes de poder controlar el robot.

2.6.1 Exportar certificado SSL

En primer lugar, debe enviar una solicitud de inicio de sesión al servidor de autenticación integrado. Si desea descargar el certificado SSL actualmente cargado, puede hacerlo exportando el certificado a través de su navegador. La siguiente ilustración muestra cómo hacerlo mediante Google Chrome. Otros navegadores admiten opciones similares.

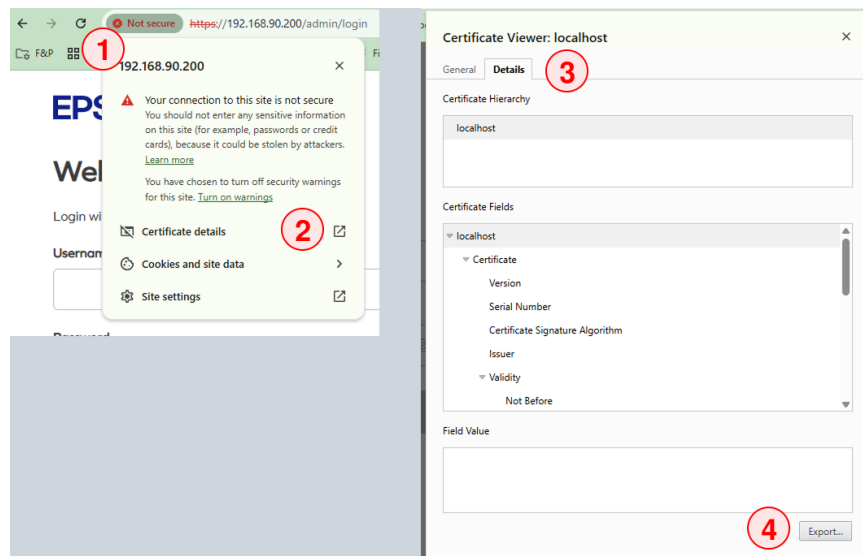


Figura 2.6.1: Exportación del certificado SSL

1. Mostrar información, 2. Acceder a los detalles del certificado, 3. En el visor de certificados, acceda a «Detalles», 4. Exportar certificado

2.6.2 Inicio de sesión del usuario

En primer lugar, debe enviar una solicitud de inicio de sesión al servidor de autenticación integrado, al que se puede acceder a través de **TCP** en el **puerto 5001**. Para ello, debe conocer la dirección IP del robot, sus datos de inicio de sesión (nombre de usuario y contraseña) y utilizar el mismo certificado SSL que el robot. A continuación se muestra un ejemplo en Python que envía la solicitud de inicio de sesión a través de un socket TCP y devuelve el token de acceso. También puede utilizar otros lenguajes de programación para realizar esta tarea.

Nota: Actualmente, el inicio de sesión sólo es posible a través de TCP. Si desea controlar su robot mediante uno de los módulos de conexión distintos de TCP (por ejemplo, ROS/REST/Modbus), deberá utilizar igualmente TCP para obtener el token de acceso. Como alternativa, puede desactivar la autenticación para estos módulos de conexión en la configuración de AX Portal. Atención: Si desactiva la autenticación, cualquier cliente podrá controlar el robot a través de ROS/REST sin necesidad de autenticarse previamente.

```
import json, socket, ssl

# define the robot ip address, login credentials, and path to SSL certificate
ip_address = "10.0.0.203"
username = "user_xyz"
password = "my_fancy_password"
session_id = "my_session_id"
ssl_cert = "/home/user/robot-ssl/cert.pem"

# create request to send
request = {"request": "authentication", "action": "login", "data":
          {"username": username, "password": password, "session_id": session_id}}

# prepare SSL context to use with the connection
ssl_context = ssl.SSLContext(ssl.PROTOCOL_TLS_CLIENT)
ssl_context.load_verify_locations(ssl_cert)
ssl_context.check_hostname = False
ssl_context.verify_mode = ssl.CERT_NONE

with socket.socket(socket.AF_INET, socket.SOCK_STREAM, 0) as sock:
    with ssl_context.wrap_socket(sock, server_hostname=ip_address) as ssock:
        # connect to the authentication socket
        ssock.connect((ip_address, 5001))
        # send request (and end character)
        message = json.dumps(request) + "\x03"
        ssock.send(message.encode("UTF-8"))
        # receive answer (before end character)
        message = ssock.recv(1024).decode("UTF-8")
        answer = json.loads(message.split("\x03")[0])

token = answer["response"]["data"]["token"]
session_id = answer["response"]["data"]["session_id"]
```

2.6.3 Registro de la conexión

Una vez que disponga de un token de acceso válido, podrá utilizarlo en la función «token_login», cuyo uso se describe detalladamente en el capítulo dedicado al módulo de conexión correspondiente.

3. Exportación a USB y actualización de la copia de seguridad

3.1 Resumen

Si no se puede acceder a la interfaz de administración aunque el controlador esté en funcionamiento, puede intentar restaurar el controlador mediante el método de actualización de recuperación.

3.2 Exportación USB

El registro del sistema y la base de datos de AXPortal se pueden exportar sin necesidad de acceder a la interfaz web mediante una memoria USB:

1. Formatee una memoria USB con exFAT, FAT32 o NTFS y asegúrese de que NO lleve el nombre de sistema de archivos LOG o RECOVERY
2. Conéctela al controlador y espere unos minutos.
3. Retire la memoria USB. Debería encontrar en el directorio raíz de la memoria un archivo ZIP que contiene la base de datos y los registros del sistema pertinentes.

3.3 Ejecutar la actualización de recuperación

Para ejecutar la actualización del software de recuperación, siga estos pasos:

1. Obtenga el archivo de actualización, que suele estar en formato RBIN. Si no dispone de un archivo de actualización, póngase en contacto con el fabricante.
2. Asegúrese de tener a mano una memoria USB que pueda utilizar para la actualización. Debe estar formateada en exFAT, FAT32 o NTFS, y el sistema de archivos de la primera partición debe estar etiquetado como «RECOVERY».
3. Cargue el archivo de actualización en el directorio raíz de la memoria USB.
4. Apague el robot y reinicielo. Espere al menos 1 minuto hasta que el robot se haya iniciado correctamente.
5. Conecte la memoria USB preparada a cualquier puerto USB del controlador.
6. Espere hasta que la actualización haya finalizado. Tenga en cuenta que esto puede tardar **hasta 15 minutos**. Intente acceder a la interfaz de administración y compruebe si se ha instalado la nueva versión.
7. **Importante:** Retire la memoria USB del controlador. Si olvida este paso, la actualización se volverá a ejecutar tras el reinicio, lo cual no es deseable.

4. Interfaz de escritorio (GUI)

4.1 Introducción

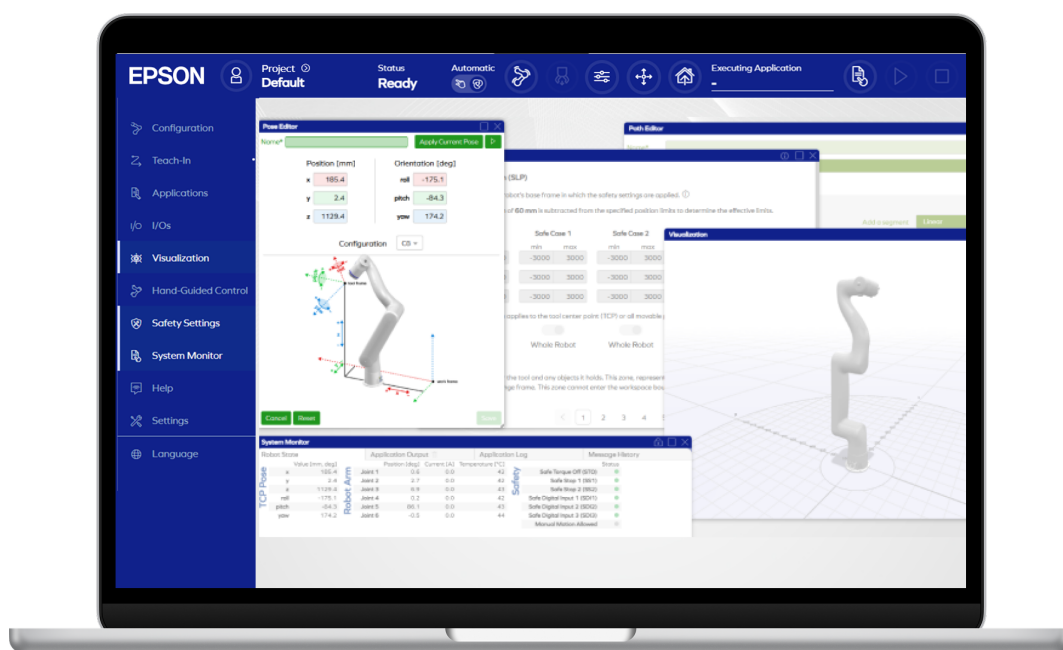


Figura 4.1.a: Ejemplo de captura de pantalla de la interfaz de escritorio

La interfaz de escritorio de AX Portal es la interfaz gráfica de usuario (GUI) más importante a través de la cual el usuario controla el robot, programa aplicaciones, edita elementos de la base de datos y visualiza el estado actual del robot. Esta GUI se ha diseñado para que resulte intuitiva y, en gran medida, autoexplicativa incluso para usuarios sin mucha experiencia en el control de robots. Puede acceder a esta interfaz a través de la interfaz de administración iniciando sesión y utilizando el botón «Cambiar a la interfaz de escritorio».

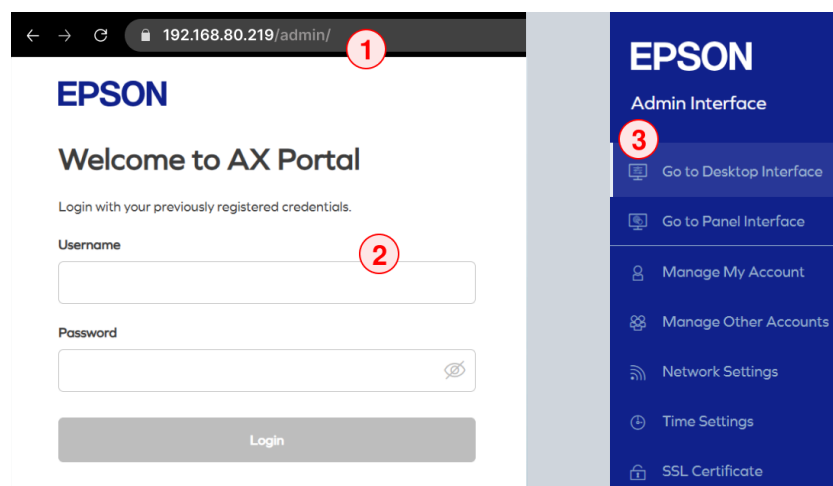


Figura 4.1.b: Acceso a la interfaz de escritorio

1. Introducir la dirección IP en la URL, 2. Introducir los datos de inicio de sesión, 3. Cambiar a la interfaz de escritorio

La interfaz de escritorio contiene todas las funciones de control disponibles para el robot, así como funciones adicionales para observar y modificar su estado. Para ejecutar fácilmente aplicaciones programadas sin tener que modificar aplicaciones o ajustes, recomendamos utilizar la interfaz de panel. En este capítulo se explican detalladamente todas las ventanas, menús y elementos de la interfaz de escritorio.

En algunos navegadores (especialmente en Windows), el escalado o la ampliación de la pantalla están configurados de forma predeterminada en un valor alto (por ejemplo, 150 %), lo que puede provocar que algunas partes de la interfaz (como Cockpit o el logotipo) aparezcan recortadas. Para obtener la mejor visualización, recomendamos ajustar la escala de visualización al 100 % (o incluso menos).

4.2 Ventana principal

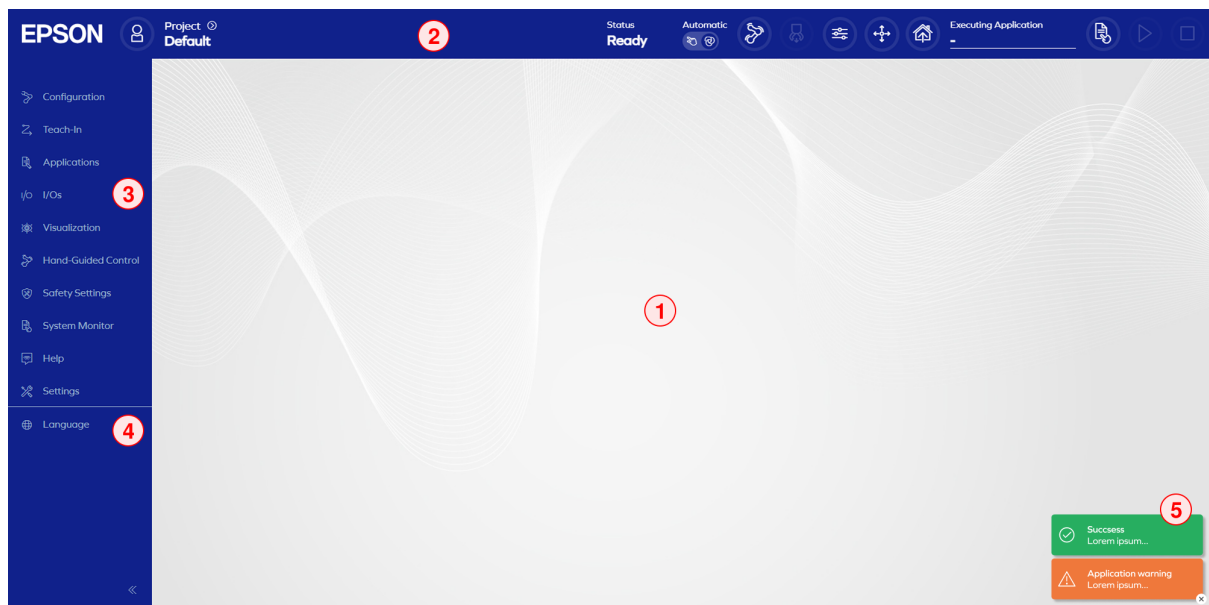


Figura 4.2: Elementos de la interfaz de escritorio

1. Área de trabajo, 2. Panel de control, 3. Menús, 4. Cambiar idioma, 5. Mensajes emergentes

Una vez que haya conectado correctamente su ordenador o dispositivo a la interfaz de escritorio, aparecerá la ventana principal en su navegador web.

Esta ventana principal contiene los siguientes elementos:

- un panel de control en la parte superior de la ventana, que muestra el estado actual del programa, los mensajes y el acceso rápido a los controles del robot,
- los botones de menú situados en el lado izquierdo de la ventana, en los que puede hacer clic para navegar por los menús disponibles (todos ellos se explican más adelante en este capítulo),
- el número de versión en la esquina inferior derecha,
- un espacio para ventanas emergentes de mensajes en el lado derecho de la pantalla y
- el área de trabajo principal, en la que el usuario puede mover y organizar los menús.

4.2.1 Resumen de menús

En la parte izquierda de la ventana principal encontrará botones para acceder a los distintos menús de AX Portal. La finalidad concreta y la información detallada de cada menú disponible se describen en los apartados siguientes. A continuación, le ofrecemos una breve descripción general de los menús disponibles:

	En el menú «Configuración», el usuario puede configurar y encender el robot.
	En el menú «Aprendizaje», el usuario puede enseñar al robot posturas, trayectorias y rejillas.
	En el menú «Aplicaciones», el usuario puede escribir, modificar y probar aplicaciones personalizadas.
	En el menú «E/S», el usuario puede supervisar las E/S y modificar las señales de salida.
	El menú «Visualización» muestra el movimiento actual del robot.
	En el menú «Control manual», el usuario puede mover manualmente las articulaciones del robot.
	En el menú «Configuración de seguridad», el responsable de seguridad puede modificar la configuración de seguridad.
	En el menú «Monitor del sistema», el usuario puede supervisar las salidas del sistema.
	El menú «Ayuda» contiene información útil sobre el programa y el sistema.
	En el menú «Configuración», el usuario puede realizar algunos ajustes básicos.
	Con el botón «Idioma», el usuario puede cambiar el idioma de visualización de las interfaces de usuario.

4.2.2 Panel de control

El panel de control de AX Portal está siempre visible en la parte superior de la interfaz gráfica de usuario y contiene la información y los controles más importantes. A continuación se explican detalladamente todos los elementos del panel de control de izquierda a derecha:

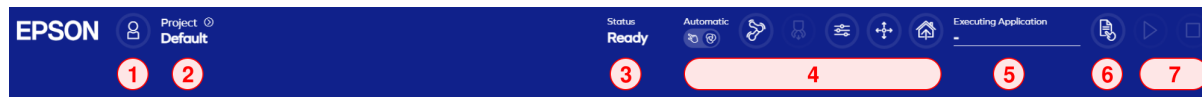


Figura 4.2.2: Panel de control

1. Opciones de la cuenta de usuario, 2. Selección de proyectos, 3. Estado del robot, 4. Controles del robot, 5. Aplicaciones actualmente en ejecución o seleccionadas, 6. Selección de aplicaciones, 7. Control principal



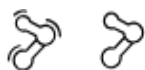
Pulse el botón «Usuario» para modificar la configuración de su cuenta o para cerrar sesión. Estas opciones le llevarán, de una forma u otra, a la interfaz de administración.

Pulse el botón «Proyecto» para acceder al menú de proyectos, donde podrá cambiar de proyecto y consultar el resumen de proyectos. Debajo de este botón verá el nombre del proyecto activo.

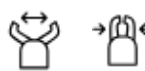
El botón «Solicitar control» solo se muestra si esta interfaz no es el control activo del robot. Encontrará más información sobre esta función de seguridad en la sección «Control activo».

La barra de mensajes de estado muestra el estado actual del programa. El estado del programa determina, en esencia, si un comando o una acción, como pulsar un botón, se puede ejecutar o no.

Con el control deslizante del modo de funcionamiento, el robot cambia entre el modo manual y el modo automático. Encontrará más información sobre esta función de seguridad en la sección **Modos de funcionamiento**.



Con el botón «Activar/desactivar control manual», el usuario puede cambiar al modo de funcionamiento manual. Cuando la función está activada, el estado cambia a «Control manual» y el usuario puede guiar el robot manualmente. El robot solo puede cambiar al modo manual si el programa se encuentra en el estado «Listo». Al desactivarlo, el estado vuelve a «Listo».



El botón «Herramienta» alterna entre las funciones «pick» y «place» de la herramienta montada. En el editor de herramientas, el usuario puede vincular estas funciones a entradas digitales. La herramienta solo se puede activar si se ha definido el comportamiento y si el estado del programa es «Listo» o «Control manual».



El botón «Control de precisión» abre una ventana para el control de precisión de los ángulos de articulación y las coordenadas del robot. Aquí, el usuario puede modificar coordenadas individuales del robot o ángulos de articulación.



El botón «Control de movimiento» abre una ventana para el control de movimiento. Aquí, el usuario puede ajustar la velocidad del robot, modificar los límites globales de velocidad y aceleración, así como cambiar los sistemas de referencia.



Con el botón «Homing», el usuario puede desplazar el robot a su posición de referencia. Esto sirve como atajo para, por ejemplo, desplazarse a una posición de inicio antes de ejecutar una aplicación o a una posición de reposo antes de apagar el sistema. La posición de referencia se puede modificar en el menú de configuración.

El área «Seleccionado actualmente» muestra qué elemento (aplicación, ruta, pose, etc.) está vinculado actualmente a los botones de control (ejecutar, pausar, continuar, detener). Esta información cambia en cuanto se hace clic en un botón de ejecución o se selecciona una aplicación.



En el menú «Selección de aplicaciones», el usuario puede seleccionar una aplicación creada previamente con el editor de aplicaciones.

-
- | | |
|--|--|
| 


 | <p>El botón «Ejecutar» ejecuta la aplicación seleccionada. El estado del programa debe ser «Listo» para reproducir una aplicación. El botón «Pausar» detiene una aplicación en ejecución. Si se pausa una aplicación durante un movimiento del robot, este se detiene suavemente. El botón «Continuar» reanuda una aplicación en pausa. Si la aplicación se ha pausado durante un movimiento, este movimiento se reanuda suavemente. El botón «Detener» detiene una aplicación en ejecución o en pausa. Si se detiene una aplicación durante un movimiento, el robot se detiene inmediatamente y mantiene su posición. Al hacerlo, el estado vuelve a «Listo».</p> |
|--|--|
-

4.2.3 Control activo (Manejo exclusivo desde una estación de control)

Para garantizar que el robot solo pueda ser controlado por una única estación de control en todo momento, las interfaces y los paneles de control están protegidos. Si usted es el primer usuario en iniciar sesión en el robot, obtendrá automáticamente el control. A esto lo denominamos «control activo». Si posteriormente otra persona (o usted mismo) inicia sesión a través de otro canal de conexión (que también puede ser otra ventana del navegador), dicha conexión no tendrá control y solo tendrá acceso para supervisar y observar al robot (acceso de lectura). A esto lo denominamos «observador pasivo».

4.2.3.1 Control activo

La estación de control activa puede compartir su control. En la interfaz, esto se hace haciendo clic en el botón «Usuarios» y seleccionando «Compartir permiso de control». Si suele trabajar con varios usuarios o interfaces en un robot, se recomienda compartir siempre el acceso de control y/o cerrar el canal de conexión.

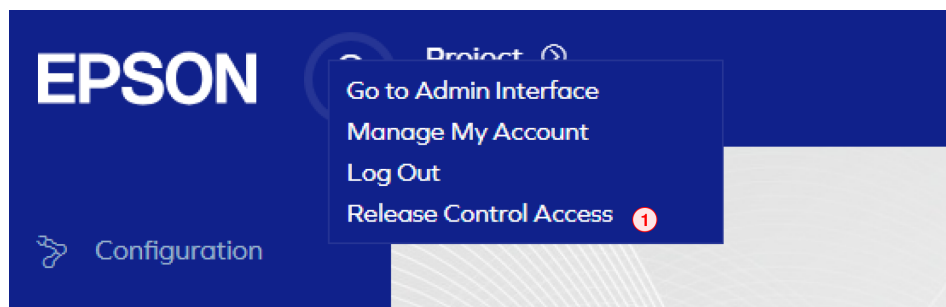


Figura 4.2.3.1 Compartir el acceso al control
1. Compartir el acceso al control

4.2.3.2 Observador pasivo

El observador pasivo puede solicitar el control. En la interfaz, esto se realiza haciendo clic en el botón «Usuario» y seleccionando «Solicitar acceso al control», o simplemente haciendo clic en el botón grande «Solicitar control» que aparece en el panel de control. Tras solicitar el control, aparece un cuadro de diálogo solicitando el acceso tanto en la estación de control activa como en el observador pasivo. Mientras que el observador pasivo solo puede esperar a que se le conceda el acceso al control, la estación de control activa puede aceptar o rechazar la solicitud.

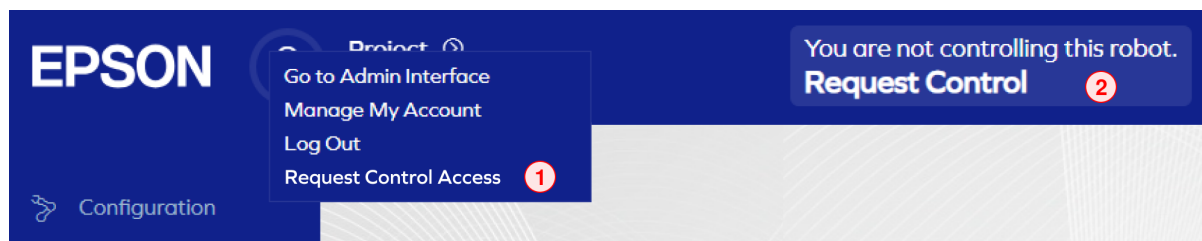


Figura 4.2.3.2 Solicitud de acceso al control

1. Solicitar acceso al control, 2. Botón «Solicitar control»

4.2.4 Estado del programa

El estado actual del programa y/o del robot es visible en el Cockpit. El estado proporciona información sobre lo que AX Portal y/o el robot están haciendo en ese momento y determina qué acciones o comandos se pueden ejecutar actualmente. Cada estado tiene sus propias características, y algunas funciones solo están disponibles en determinados estados del programa. A continuación se incluye una lista de los distintos estados y sus explicaciones:

Fuera de servicio	Los motores no reciben alimentación y los dispositivos no están conectados. Este estado se utiliza para configurar el robot antes de encenderlo. Al utilizar el control deslizante de encendido, el estado cambia a «Conectando». Otras acciones pueden cambiar el estado a «Cargando», especialmente si no se debe interrumpir una acción y esta requiere cierto tiempo para completarse.
Cargando	El robot está procesando datos, por ejemplo, está cargando una base de datos o ajustes. Una vez cargados los datos, el estado vuelve a «Fuera de servicio».
Conectando	El robot está encendiendo sus motores y conectando sus dispositivos. Una vez establecida la conexión, el robot comienza a controlar activamente sus motores y el estado cambia a «Listo».
Listo	El programa está listo para recibir comandos y/o ejecutar acciones, como por ejemplo, ejecutar una aplicación, coger y soltar objetos, etc. Este es el estado de inactividad del robot.
En ejecución	Actualmente se está ejecutando una aplicación o un movimiento. Esta aplicación o movimiento se puede pausar o detener con el botón correspondiente «Pausar» o «Detener». Una vez que la aplicación llega al final de su código o finaliza el movimiento en curso, el estado vuelve a «Listo».
En pausa	Actualmente hay una aplicación o movimiento en pausa. Pulse el botón «Continuar» para reanudar, con lo que el estado volverá a «En ejecución», o pulse el botón «Detener» para cancelar, con lo que el estado volverá a «Listo».

Control manual	El control manual (HGC) está activado para una o varias articulaciones del robot. En este modo, puede guiar manualmente al robot a cualquier posición ejerciendo fuerza sobre sus articulaciones. Al mismo tiempo, el robot mantiene su propio peso compensando activamente la gravedad. Tenga en cuenta que debe pulsar el botón HGC o accionar el pulsador de confirmación para mover el robot. Desactive de nuevo el modo de control manual para restablecer el estado a «Listo».
En proceso	Se está ejecutando una acción que requiere tiempo, como guardar/cargar datos o calcular una trayectoria. Una vez completada con éxito esta acción, el estado volverá a su valor anterior.
Detenido	Normalmente, el estado «Parada» se restablece automáticamente a «Listo» en muy poco tiempo. Si no fuera así, es posible que una parte de AX Portal no se haya completado correctamente. Lamentablemente, esto solo se puede solucionar reiniciando el robot.
Parada de emergencia Parada normal Parada de seguridad Colisión Parada de seguridad Limitación de posición límite de posición	Estos estados corresponden a diferentes interrupciones. Si se activa una interrupción, se muestra el estado correspondiente y se abre la ventana de interrupciones para gestionar la interrupción. Una vez resuelta la interrupción, el estado suele cambiar a «Se reanuda».
Se reanuda	Una vez resuelta una interrupción, el estado cambia a «Se reanuda». Esto requiere que el usuario confirme manualmente que es seguro continuar con la aplicación o el movimiento del robot.
Error	Se ha producido un error que no se ha podido resolver automáticamente o que requiere una atención especial por parte del usuario. Consulte el mensaje de error e intente resolverlo si es posible.
Desconexión	El robot está apagando sus motores, desconectando sus dispositivos y cerrando todos los módulos en ejecución. A continuación, el estado cambia a «Fuera de servicio».

4.2.5 Modos de funcionamiento

El robot de la serie AX dispone de dos modos de funcionamiento: modo manual y modo automático. El modo de funcionamiento se puede cambiar mediante el control deslizante correspondiente en la cabina.

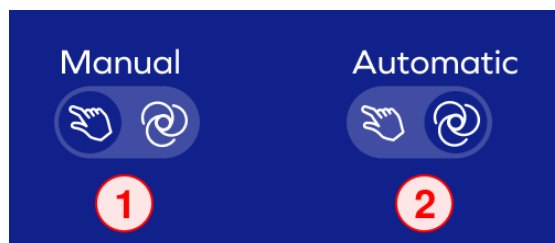


Figura 4.2.5 Interruptor para seleccionar el modo de funcionamiento

 ADVERTENCIA	El modo automático sólo debe utilizarse si el sistema está equipado con las funciones de seguridad necesarias, ha sido comprobado y su seguridad ha sido confirmada mediante una evaluación de riesgos. El uso del modo automático en otras situaciones es muy peligroso. Encontrará más información en el «AX6 / RC-A101: Manual de seguridad».
---	---

4.2.5.1 Modo automático

El modo automático se utiliza habitualmente durante el funcionamiento normal, es decir, cuando se ha comprobado que las aplicaciones funcionan correctamente. Este modo de funcionamiento permite utilizar todo el rango de velocidades del robot.

4.2.5.2 Modo manual

El modo manual es una función de seguridad y se utiliza habitualmente al modificar las aplicaciones del robot, especialmente al programar nuevas posiciones o al probar cambios en las aplicaciones. En el modo manual, el robot se mueve a velocidad reducida (máx. 250 mm/s) y solo cuando se acciona el interruptor de autorización. Se trata de un interruptor de seguridad de 3 posiciones. Si no se acciona el interruptor de autorización, se informa al usuario mediante una advertencia que indica que no se permiten movimientos.



Figura 4.2.5.2: Ejemplo de un interruptor de autorización

4.2.6 Control de precisión

Pulse el botón de control de precisión en el panel de control para abrir los controles de precisión. Aquí se puede ajustar con precisión el robot con una exactitud de hasta 0,1 [mm]/[°]. Esta ventana cuenta con dos pestañas. La primera pestaña permite al usuario modificar las coordenadas euclidianas actuales del TCP (punto central de la herramienta), lo que corresponde al sistema de referencia de la herramienta actual expresado en el sistema de referencia de trabajo. La segunda pestaña permite al usuario modificar los ángulos de articulación actuales del robot.

Estas pestañas cuentan con las siguientes funciones:

Campos de entrada	Al introducir un número en el campo de entrada, se modifica una coordenada individual o un valor de articulación.
Control deslizante	El control deslizante define tanto el paso, que determina la distancia que recorre el robot al hacer clic brevemente en un botón de control, como la velocidad máxima de movimiento, que determina la rapidez con la que se desplaza el robot con estos controles.
Botones	Al pulsar brevemente un botón, se aplica el paso seleccionado en la dirección indicada por el botón. Por el contrario, al mantener pulsado un botón, se inicia un movimiento continuo en la dirección indicada.
Configuración	Se puede alcanzar un TCP determinado mediante hasta ocho combinaciones de ángulos de articulación. Una combinación de ángulos de articulación se denomina «configuración» de dicha pose. Al cambiar el valor «Configuración» en la lista desplegable, el usuario puede seleccionar una configuración específica para la pose actual. Y al pulsar el botón «Siguiente configuración», el usuario puede recorrer las configuraciones disponibles de la pose actual. Tenga en cuenta que cambiar la configuración no es un paso de ajuste fino; al hacerlo, se mueven todas las articulaciones.
Guardar como pose	Al pulsar el botón «Guardar como pose», esta pose se guarda con un nombre nuevo y único. Todas las poses disponibles se enumeran en el menú «Aprendizaje».

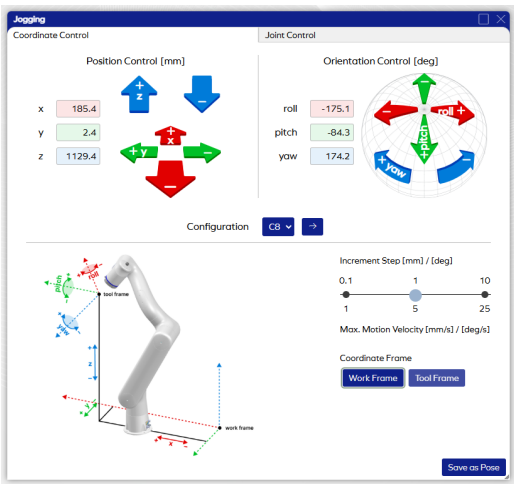


Figura 4.2.6.a: Control de precisión – Control de coordenadas

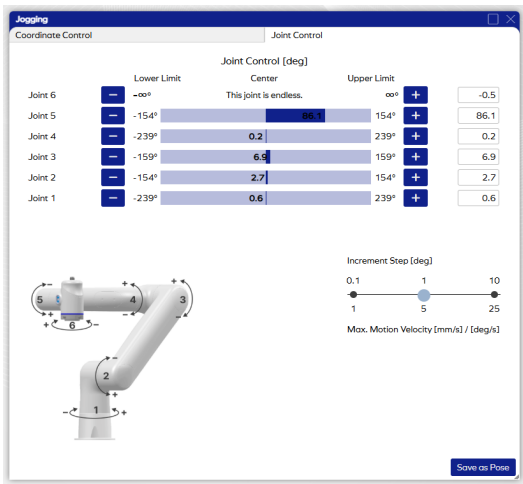


Figura 4.2.6.b: Control de precisión – Control de articulaciones

4.2.7 Control de movimiento

Pulse el botón «Control de movimiento» en el panel de control para abrir la ventana de control de movimiento. Esta ventana cuenta con las tres pestañas siguientes:

4.2.7.1 Límites de movimiento

La pestaña «Límites de movimiento» contiene ajustes para limitar la velocidad máxima del robot y para reducir la velocidad de todos los movimientos del robot mediante un factor de escala. Estos controles resultan útiles durante la puesta en marcha de aplicaciones robóticas, por ejemplo, para limitar la velocidad al validar una nueva aplicación o al modificar movimientos en aplicaciones existentes.

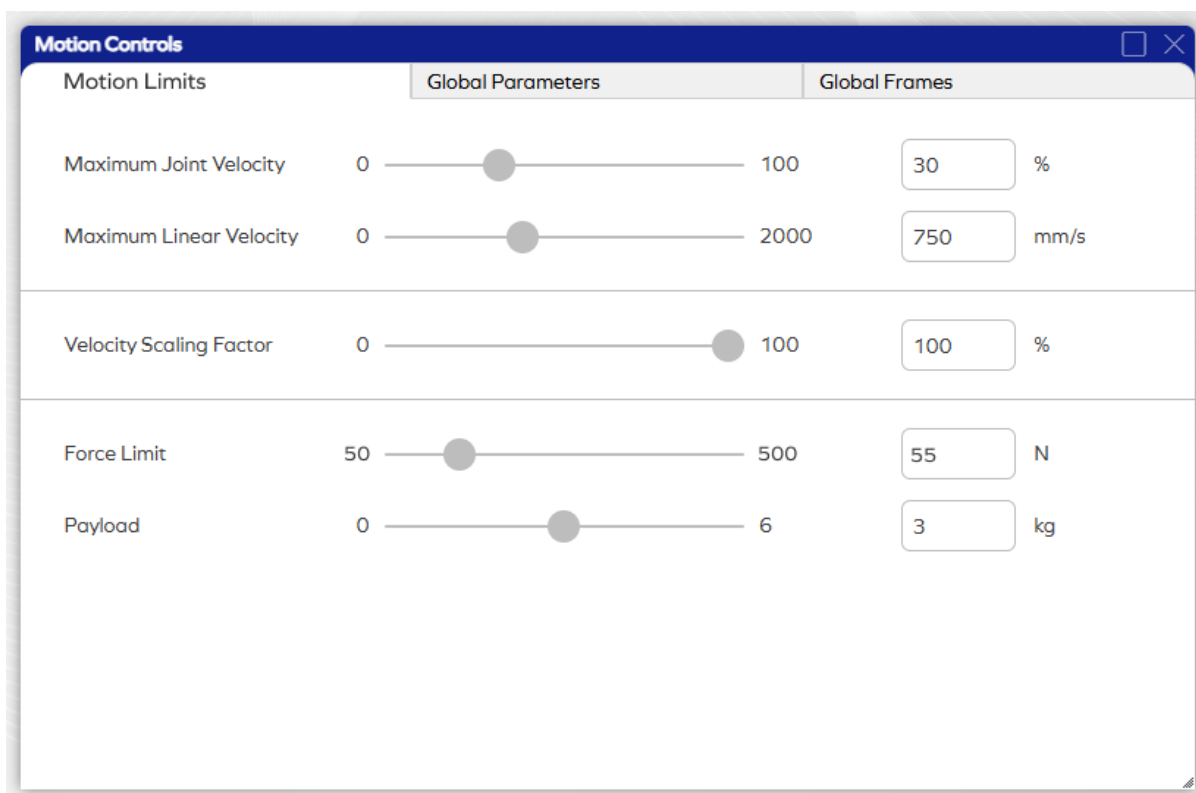


Figura 4.2.7.1: Control de movimiento – Pestaña «Límites de movimiento»

Por su parte, el límite de fuerza y la carga útil se utilizan para ajustar la sensibilidad a las colisiones del robot. El límite de fuerza define un límite máximo para las fuerzas externas admisibles en la herramienta, y la carga útil define el peso total que soporta el robot (incluida su herramienta).

4.2.7.2 Parámetros globales

La pestaña «Parámetros globales» contiene ajustes para definir la velocidad y la aceleración predeterminadas para diferentes comandos de movimiento. Los comandos de movimiento se dividen en las siguientes categorías:

- Los comandos para mover el robot de forma óptima en cuanto al tiempo utilizan los argumentos «joint_velocity» y «joint_acceleration», cuyos valores predeterminados se establecen aquí como **Velocidad global de la articulación** y **Aceleración global de la articulación**.
- Los comandos para mover el robot en una dirección o forma determinada (lineal, circular) utilizan los argumentos «linear_velocity» y «linear_acceleration», cuyos valores predeterminados se establecen aquí como **Velocidad lineal global** y **Aceleración lineal global**.
- Los comandos para mover el robot de tal manera que el TCP (punto central de la herramienta) permanezca en la misma posición y solo cambie su orientación utilizan los argumentos «angular_velocity» y «angular_acceleration», cuyos valores predeterminados se establecen aquí como **velocidad de rotación global** y **aceleración de rotación global**.
- Los comandos para desplazarse a lo largo de una trayectoria que contiene varios segmentos de movimiento utilizan el argumento «blend_radius», cuyo valor predeterminado se establece aquí como **Radio de transición global**. Este parámetro se utiliza para suavizar la interpolación entre segmentos de la trayectoria.

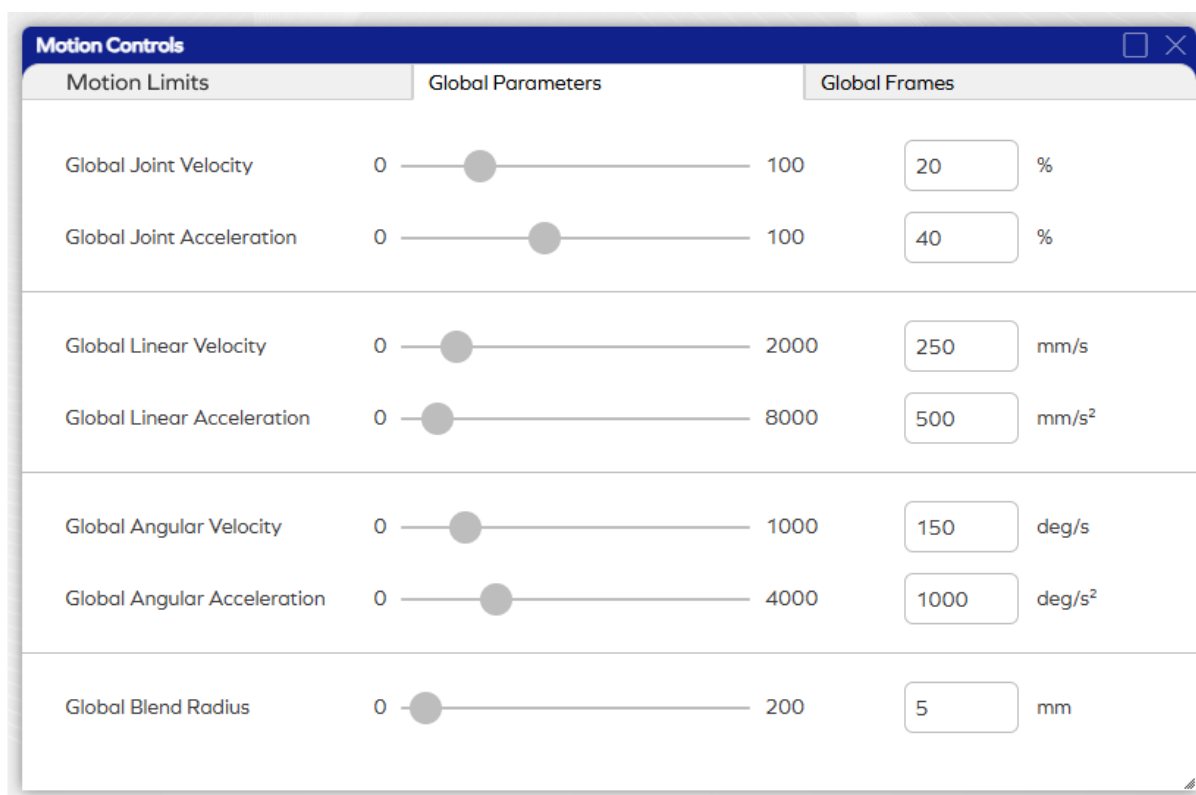


Figura 4.2.7.2: Control de movimiento – Pestaña «Parámetros globales»

4.2.7.3 Sistemas de coordenadas globales

La pestaña «Sistemas de coordenadas globales» contiene ajustes para definir los sistemas de coordenadas de referencia predeterminados del robot. Mientras que el sistema base y el sistema de brida están fijados al robot, el sistema de trabajo y el sistema de herramienta pueden definirse de forma relativa a estos sistemas de referencia fijos:

- El **sistema base** está fijado al centro de la superficie de montaje inferior del robot, de modo que el eje z apunta hacia el brazo del robot y el eje x se aleja de los cables conectados.
- El **sistema de bridas** está fijado al centro de la placa de montaje superior del robot. Su orientación coincide con el sistema de referencia del robot cuando todas las articulaciones del robot se encuentran en cero grados.
- El **sistema de trabajo** es el sistema de referencia en el que el usuario expresa las coordenadas del TCP (punto central de la herramienta) en las aplicaciones. Por defecto, el sistema de trabajo coincide con el sistema base del robot. Muchos comandos de movimiento utilizan el argumento «work_frame», cuyo valor por defecto se define aquí como sistema de trabajo global.
- El **sistema de herramienta** define la posición y la orientación del TCP (punto central de la herramienta), expresadas en el sistema de brida. Por defecto, el sistema de trabajo coincide con el sistema de brida del robot. Muchos comandos de movimiento utilizan el argumento «tool_frame», cuyo valor por defecto se establece aquí como sistema de herramienta global.

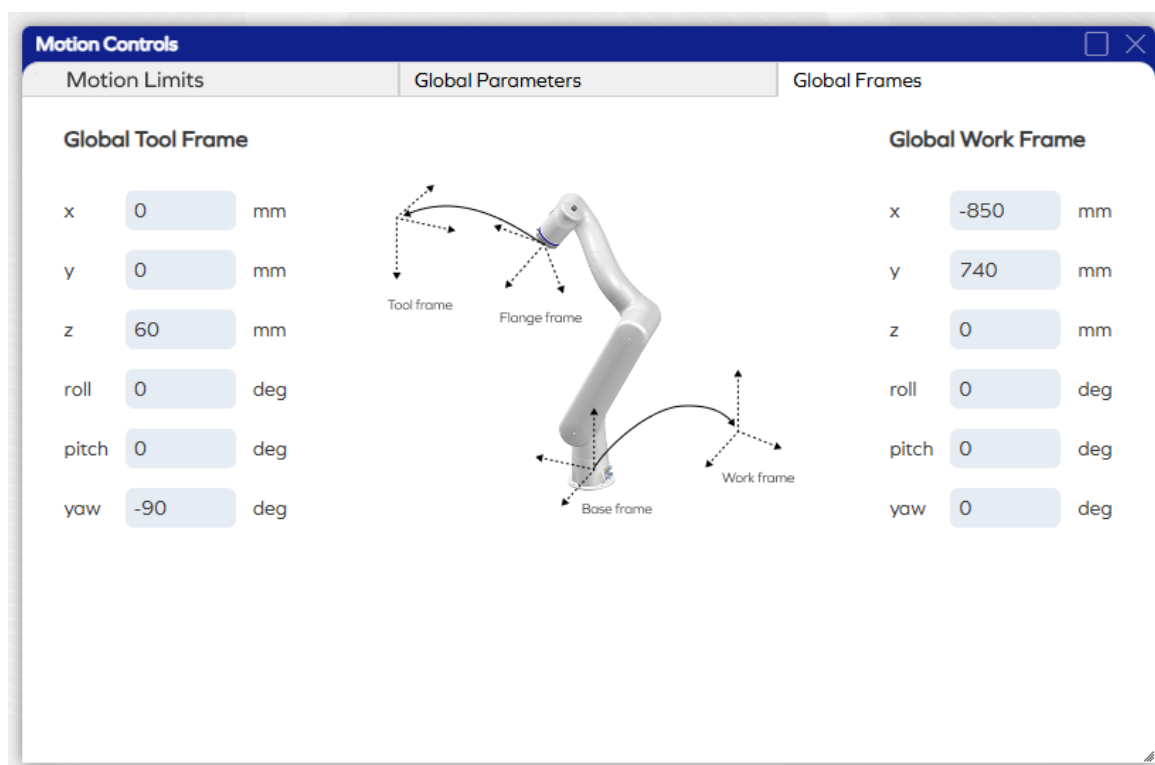


Figura 4.2.7.3: Control de movimiento – Pestaña «Sistemas de coordenadas globales»

4.3 Bloques y base de datos

Para permitir una programación de aplicaciones sencilla e intuitiva, AX Portal ofrece una estructura orientada a objetos con diversos bloques de construcción. Estos bloques de construcción (en lo sucesivo, «elementos») pueden crearse individualmente en distintos editores, a los que se puede acceder a través de la interfaz de escritorio. AX Portal permite al usuario vincular estos elementos entre sí de forma inteligente para crear aplicaciones orientadas a objetos, lo que da como resultado un código de aplicación breve, fácil de leer y que, por lo general, no contiene muchos números ni coordenadas.

A continuación se incluye una lista de los elementos disponibles:

	Las aplicaciones contienen principalmente el código de programa (y/o bloques) que se va a ejecutar y la configuración específica de la aplicación. El código de programa es compatible con Python 3.13 y, además de las funciones básicas de Python, incluye muchas funciones de la API (Interfaz de Programación de Aplicaciones) interna de control de robots. Las aplicaciones se crean y modifican con el editor de aplicaciones.
	Las poses son posiciones del punto central de la herramienta (TCP) con orientaciones específicas, definidas de forma unívoca mediante una combinación de ángulos de articulación. Las poses se crean y modifican con el editor de poses.
	Las trayectorias son secuencias de poses que describen movimientos suaves del TCP y, por lo tanto, movimientos de todo el robot. Las trayectorias se crean y modifican con el editor de trayectorias.
	Las rejillas son disposiciones de puntos personalizables, de hasta tres dimensiones, que definen, por ejemplo, las posiciones de bandejas predefinidas. Los puntos de la rejilla se extraen y se utilizan como iteradores en las aplicaciones. Las rejillas se crean y modifican con el editor de rejillas.
	Las E/S se dividen en entradas y salidas digitales y analógicas. Se pueden configurar a través del menú «Configuración de E/S», y el valor de las E/S es visible en el menú «E/S». Las E/S sirven para la conexión y la comunicación con dispositivos externos.

Todos estos elementos se almacenan en la base de datos de AX Portal. Puede descargar la base de datos de su robot en cualquier momento (para disponer de una copia de seguridad de toda su configuración) o cargar una nueva base de datos (lo que sobrescribirá la base de datos actual y todos sus elementos) a través del menú de configuración. Además, el menú de proyectos ofrece una clara visión general de todos los proyectos y elementos almacenados en su base de datos.

4.4 Menú de proyectos

Haga clic en el nombre del proyecto seleccionado actualmente en el panel de control para acceder al menú de proyectos. Este menú contiene una vista general de los elementos disponibles, como poses, trayectorias, rejillas y aplicaciones. Estos elementos se agrupan en varios proyectos. Solo se selecciona un proyecto a la vez, y solo los elementos del proyecto seleccionado son visibles, editables y utilizables en los demás menús de AX Portal.

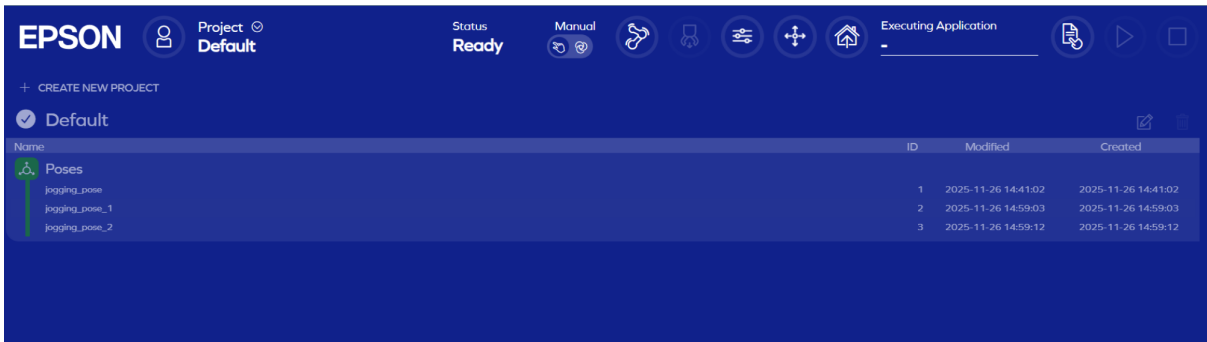


Figura 4.4: Menú de proyectos

-
-  Pulse el botón «Crear nuevo proyecto» para crear un nuevo proyecto vacío.
 -  Pulse el círculo situado a la izquierda del nombre del proyecto para seleccionar dicho proyecto.
 -  Pulse el botón «Copiar» de un elemento del proyecto para crear una copia exacta de dicho elemento dentro del mismo proyecto. El nombre del elemento copiado se modifica ligeramente y termina con «(copia)».
 -  Pulse el botón «>>» situado a la derecha del nombre del proyecto para cambiar el nombre de este proyecto. Pulse el botón «Editar» de un elemento del proyecto para abrir el editor correspondiente y poder verlo o editarlo.
 -  Pulse el botón «Eliminar» situado a la derecha del nombre del proyecto para eliminar dicho proyecto, incluidos todos sus elementos. Pulse el botón «Eliminar» de un elemento del proyecto para eliminar ese elemento concreto del proyecto.

Haga clic en el icono de un elemento (por ejemplo, poses, trazados, etc.) para mostrar u ocultar la lista de elementos de esa sección concreta.

Mueva un elemento de un proyecto a otro arrastrándolo con el ratón desde el proyecto original y soltándolo en el proyecto de destino. Mantenga pulsada la tecla CTRL mientras arrastra el elemento para crear una copia de este en el proyecto de destino.

4.5 Menú de configuración

Pulse el botón «Configuración» en la ventana principal para acceder al menú de configuración. En este menú puede configurar la base de datos y la configuración del hardware (E/S y herramientas). Si utiliza un robot estacionario para una aplicación concreta, solo tendrá que configurar el sistema una vez y, a partir de entonces, utilizar esta ventana únicamente para encender el robot. Incluso puede automatizar la puesta en marcha del robot y la ejecución de una aplicación mediante los ajustes correspondientes del menú de configuración. Sin embargo, si utiliza su robot para múltiples fines, recurrirá a este menú con mayor frecuencia y utilizará diferentes herramientas o incluso diferentes bases de datos para distintos proyectos y fines.

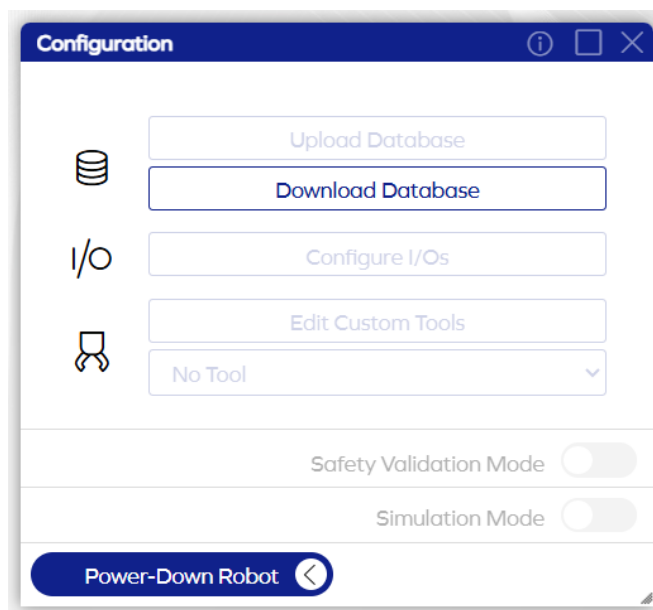


Figura 4.5: Menú de configuración

4.5.1 Base de datos (descarga y carga)

En la sección «Base de datos» puede descargar la versión actual de la base de datos de su aplicación o sustituir (sobrescribir) la base de datos por otra versión compatible. Al descargar la base de datos, se crea una copia local del archivo ZIP de la base de datos en su dispositivo. Se recomienda realizar copias de seguridad de su base de datos con regularidad, especialmente si tiene muchas aplicaciones y ha invertido mucho tiempo en escribir y perfeccionar sus aplicaciones. Al cargar una base de datos, se le pedirá que seleccione un archivo ZIP de su dispositivo. Al seleccionar el archivo de la base de datos, la base de datos actual se sobrescribirá y se sustituirá por la cargada.

Al descargar una base de datos en un robot y cargarla en otro, puede transferir todas las aplicaciones y elementos entre robots. Se recomienda realizar copias de seguridad de la base de datos con regularidad.

Tenga en cuenta, además, que las bases de datos tienen versiones y que no todas las bases de datos son compatibles con todas las versiones de AX Portal. Las bases de datos más antiguas se pueden importar a versiones más recientes del software, pero las versiones más recientes de la base de datos no se pueden cargar en versiones más antiguas del software. En otras palabras: las bases de datos no son compatibles con versiones anteriores.

4.5.2 Configuración de E/S

Pulse el botón «Configurar E/S» para abrir la ventana «Configuración de E/S». Aquí puede configurar todas las E/S, asignar funciones predefinidas a las entradas digitales y establecer nombres personalizados para sus E/S. Los nombres definidos por el usuario pueden utilizarse en las aplicaciones para mejorar la legibilidad de su código de programa. Allí puede sustituir fácilmente los identificadores enteros (int) por los nombres de E/S definidos (str).

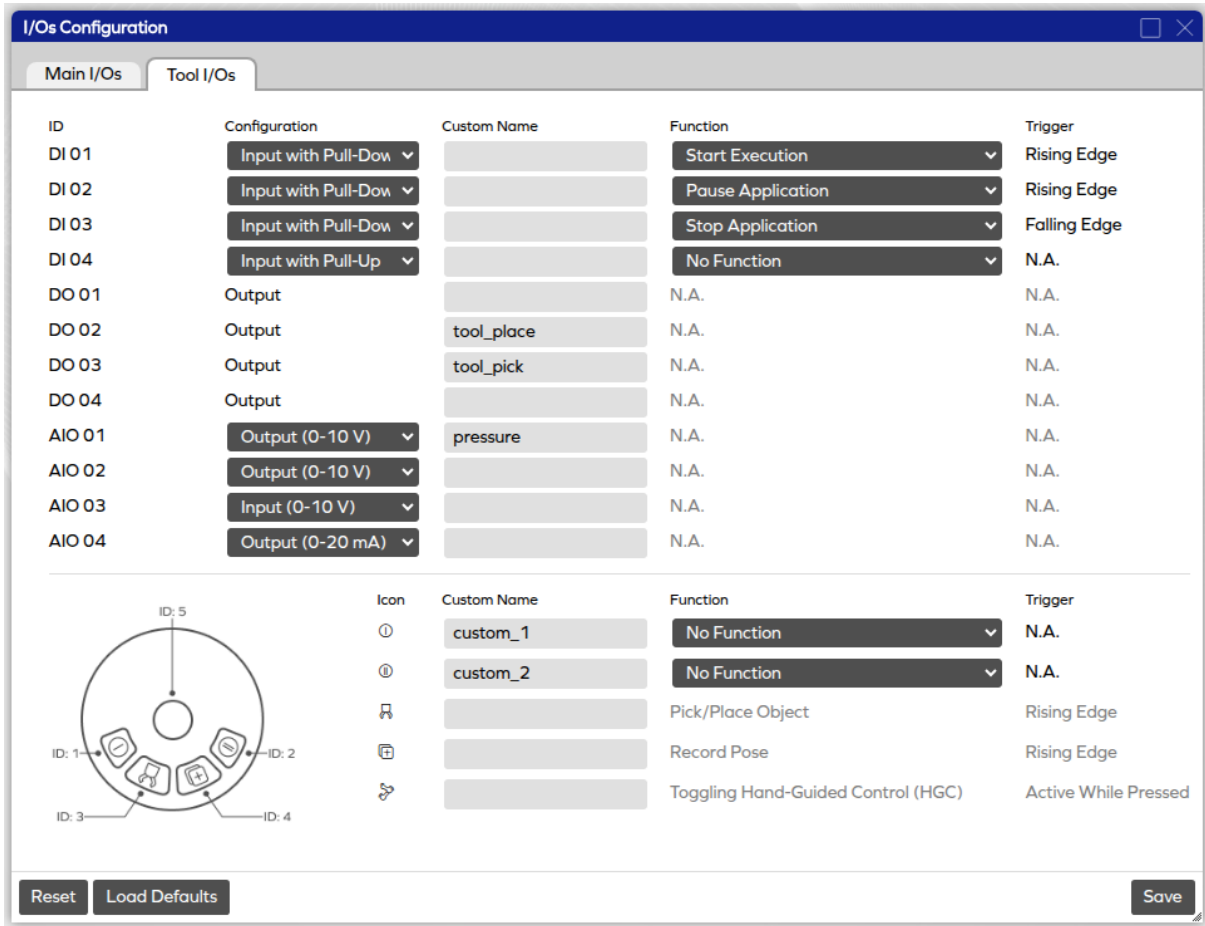


Figura 4.5.2.a: Menú «Configuración de E/S»

La serie AX se suministra con una «interfaz de botones» situada en el eje superior. A los dos botones personalizados se les pueden asignar funciones exactamente de la misma manera que en este menú se asignan funciones a otras entradas digitales.

Tenga en cuenta que, al activar una función a través de una entrada digital, se retira el control activo de la estación de control actual.

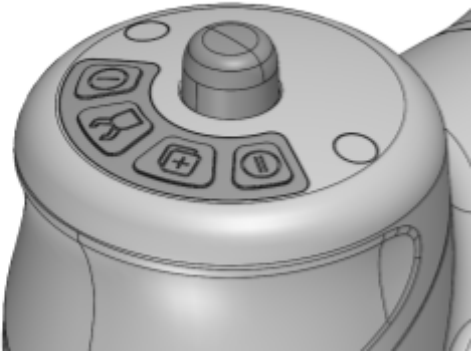


Figura 4.5.2.b: Interfaz de botones AX

Están disponibles las siguientes funciones predefinidas para entradas digitales, que puede configurar para que se activen en el flanco ascendente (por defecto) o en el flanco descendente:

Sin función	No se activa nada. Este es el ajuste predeterminado.
Pausar la aplicación	Pausa de la aplicación que se está ejecutando actualmente. Esto equivale a pulsar el botón «Pausar» del panel de control.
Cancelar la aplicación	Detener la aplicación que se está ejecutando actualmente o que está en pausa. Esto equivale a pulsar el botón «Detener» del Cockpit.
Iniciar ejecución	Ejecutar la aplicación seleccionada actualmente. Esto equivale a pulsar el botón «ejecutar» del Cockpit. Cuando el robot se recupera tras una interrupción, esta función lo devuelve a la posición en la que se interrumpió y continúa con el movimiento anterior. Esto equivale a pulsar el botón «Continuar» en el cuadro de diálogo que aparece en la interfaz gráfica de usuario tras una interrupción.
Restablecimiento de seguridad	Restablecimiento de la seguridad tras una interrupción. Esto equivale a restablecer la seguridad a través del cuadro de diálogo que aparece en la interfaz gráfica de usuario tras una interrupción.
Activar el control manual	Activación del modo de control manual (HGC) para todos los ejes (mientras el estado es «listo»).
Desactivar el control manual	Desactivación del modo de control manual (HGC) para todos los ejes (mientras el estado es «control manual»).
Cambiar el control manual	Alternar entre activar y desactivar el HGC para todos los ejes. La función activada depende del estado actual.
Accionar la herramienta (cancelar)	Activación de la función de recogida de la herramienta (predefinida o vinculada a salidas digitales). Esto equivale a pulsar el botón «Retirar» del panel de control.
Manejo de la herramienta (colocar)	Activación de la función de colocación de la herramienta (predefinida o vinculada a salidas digitales). Esto equivale a pulsar el botón «colocar» del panel de control.
Manejo de la herramienta (recoger/colocar)	Cambio entre la recogida y la colocación de un objeto con la herramienta. Depende del estado actual de la herramienta.
Guardar pose	Registre la postura actual del robot y guárdela como «button_pose» (o similar).
Cambiar al modo manual	Cambio del robot al modo manual. Esto equivale al cambio de modo de funcionamiento mediante el interruptor de la cabina.

Cambiar al modo automático	Cambio del robot al modo automático. Esto equivale al cambio de modo de funcionamiento mediante el interruptor de la cabina.
-----------------------------------	--

4.5.3 Configuración de la herramienta

En el área de herramientas puede configurar la herramienta del robot. Si desea integrar una herramienta definida por el usuario, por ejemplo, una herramienta de un proveedor externo o su propia herramienta especial para una aplicación concreta, puede utilizar la opción «Editar herramientas definidas por el usuario». Al hacerlo, se abrirá la lista de herramientas disponibles con las siguientes opciones:

	Pulse el botón «Crear nueva herramienta» para abrir el editor de herramientas y definir las propiedades cinemáticas y dinámicas, así como el disparador de recogida y colocación de su herramienta.
	Pulse el botón «Filtrar/Buscar» para filtrar la lista de herramientas personalizadas por nombre. Al introducir tres o más caracteres se activa el filtro de búsqueda.
	Pulse el botón «Editar» (o haga doble clic en cualquier parte del elemento) para abrir el editor de herramientas en una ventana independiente para su visualización o edición.
	Pulse el botón «Eliminar» para eliminar de forma irreversible una herramienta de su base de datos.

4.5.4 Editor de herramientas

El editor de herramientas se divide en cuatro secciones (véase la ilustración a continuación). Para completar estos valores, es posible que tenga que consultar la ficha técnica de la herramienta que desea integrar en AX Portal. Las propiedades físicas y eléctricas de la herramienta deberían estar descritas allí.

En la primera sección, debe definir el TCP (Tool Center Point), es decir, la posición en el espacio en la que interactúa la herramienta. En el caso de una herramienta que sujeta objetos, esto significa que el TCP define el punto en el que se encuentra la herramienta durante la sujeción. El TCP se define en el sistema de coordenadas de referencia, que se sitúa exactamente en el centro de la placa de conexión superior y está orientado de la misma manera que el sistema base (siempre que todos los ejes estén a cero grados). Este sistema de referencia es el sistema de herramienta predeterminado si no se ha definido ningún TCP (x, y, z, roll, pitch y yaw, todos a cero). Desde allí, puede desplazar y girar el TCP de modo que coincida con el punto de interacción de su herramienta.

En el siguiente paso, es necesario ajustar las propiedades dinámicas de su herramienta. La calidad de funciones como la compensación de la gravedad y la detección de colisiones depende de que el modelo dinámico del robot sea lo más preciso posible; por lo tanto, merece la pena medir y definir estas propiedades de su herramienta. Debe definir la masa total de la herramienta, así como el centro de masa, que (al igual que el TCP) se define en el sistema de referencia estándar de herramientas del robot. El momento de inercia de la masa puede introducirse manualmente (si se conoce) o aproximarse automáticamente (AX Portal desconoce la forma y la distribución de masa de su herramienta).

Tool Editor : pneumatic_gripper

Name* pneumatic_gripper

1. Defining the Global Tool Frame

Translation [mm]

x 0

y 0

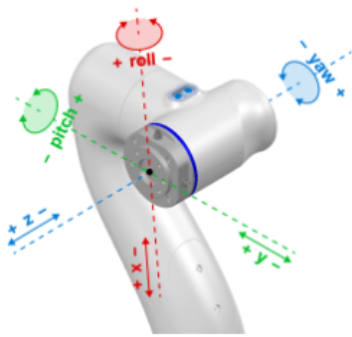
z 36

Rotation [deg]

roll 0

pitch 0

yaw 90



2. Defining the Dynamic Properties

Mass [kg] 0.662

Center of Mass [mm]

x 2 y -1.5 z 14

Rotational Moment of Inertia [kg m²] Automatic ☒ Manual

0.0004649	0	0
0	0.0004649	0
0	0	0.0005880

3. Defining the Picking Behaviour

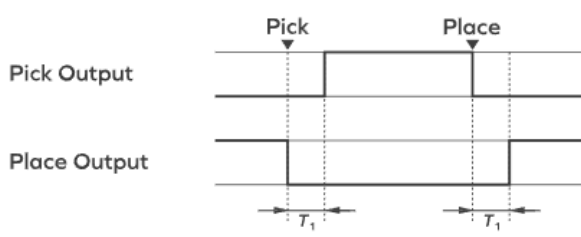
Mode Dual Output Mode

Select Digital Output for Picking tool DO 03 (tool_pick)

Select Digital Output for Placing tool DO 02 (tool_place)

Output State for Picking & Placing High

Delay Between Output Triggers T_1 [ms] 20



Pick Output

Place Output

Pick

Place

T_1

Cancel Reset Save & Close

Figura 4.5.4: Editor de herramientas

Si su herramienta dispone de un mecanismo para agarrar un objeto, aquí puede especificar qué salidas digitales se activan mediante las funciones «levantar» y «colocar», a las que se puede acceder tanto desde el panel de control («Cockpit») como a través de las funciones de programación. Están disponibles los siguientes comportamientos de agarre:

4.5.4.1 Modo de salida única

Este es el comportamiento de agarre más sencillo de los disponibles. Basta con definir una salida digital que se utilice para alternar entre agarrar y soltar, y puede especificar si el agarre se activa con una señal HIGH o LOW.



Figura 4.5.4.1: Modo de salida única con ALTO para recoger (y BAJO para soltar)

4.5.4.2 Modo de salida doble

Este comportamiento de agarre le permite definir salidas digitales independientes para la recogida y la colocación. Al cambiar el estado de la herramienta, ambas salidas se activan sucesivamente con un breve retraso T_1 (en milisegundos). Además, puede especificar si el estado activo se refiere a una señal HIGH o LOW.

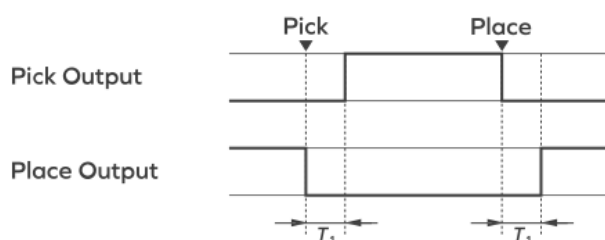


Figura 4.5.4.2: Modo de doble salida con HIGH como estado de activación activo

4.5.4.3 Modo de activación separada

En el modo de activación separada, puede definir dos salidas digitales: una que determina la dirección (si se debe recoger o colocar) y otra que determina el disparador (si se debe activar la dirección seleccionada). Para la salida de dirección, puede elegir si se utiliza una señal HIGH o LOW para recoger (la otra es para colocar). Para la salida de activación, también puede elegir si se utiliza una señal HIGH o LOW para la activación. Además, puede definir el breve retraso T_1 (en milisegundos) entre la selección de la dirección y la activación de la dirección seleccionada, así como la breve duración T_2 (en milisegundos) que determina cuánto tiempo dura la activación.

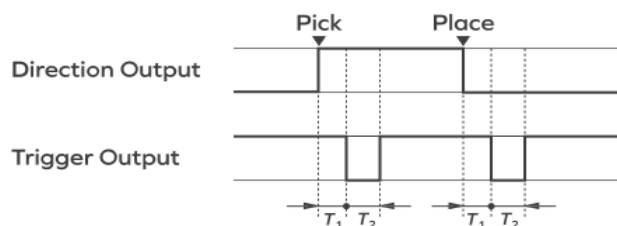


Figura 4.5.4.3: Modo de activación independiente con HIGH para el agarre y LOW como activador

4.5.5 Puesta en marcha del robot

Una vez cargada la base de datos y configuradas las E/S y la herramienta, es el momento de encender el robot. Para ello, deslice el control deslizante «Poner en marcha el robot» completamente hacia la derecha, lo que iniciará la puesta en marcha.

Si no desea encender el hardware, sino sólo modificar el contenido de la base de datos (aplicaciones, poses, etc.) sin utilizar el hardware (por ejemplo, sin mover el robot, sin leer ni escribir en las E/S), puede activar previamente el interruptor «Modo de simulación» y, a continuación, deslizar el control deslizante «Poner en marcha el robot» hacia la derecha. Tenga en cuenta que, en el modo de simulación, todos los movimientos se simulan en tiempo real.

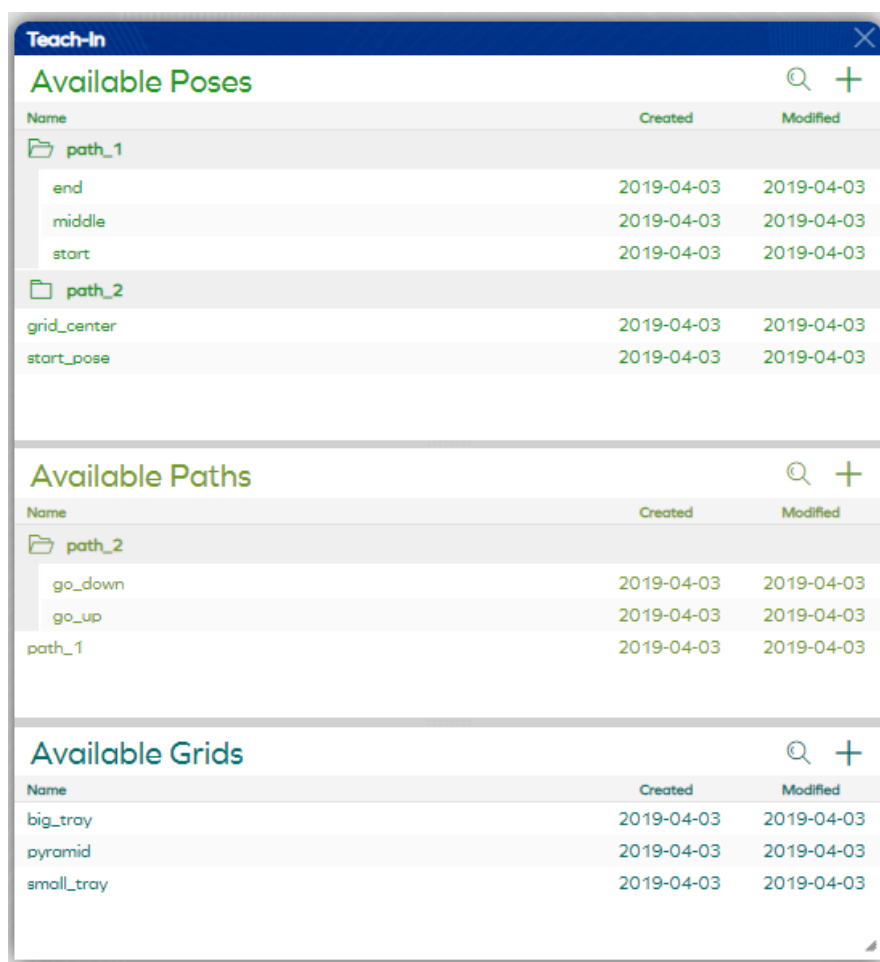
Mientras el robot está encendido, no puede modificar ninguno de los ajustes de hardware del robot. Solo puede descargar la base de datos o deslizar el control deslizante «Poner el robot fuera de servicio» hacia la izquierda, lo cual se recomienda cuando el robot está inactivo o en modo de espera.

Si ha iniciado sesión como responsable de seguridad, aquí tendrá acceso al botón «Modo de prueba de seguridad». La activación de este modo indica al software que ignore los límites de posición, velocidad y fuerza, lo que puede utilizarse para comprobar si el dispositivo de seguridad se activa realmente en caso de que se supere un límite. Tenga en cuenta que el uso de cualquiera de los controles de movimiento en este modo seguirá anulando los límites.

4.6 Menú de aprendizaje

Pulse el botón «Aprendizaje» en la ventana principal para acceder al menú de aprendizaje. Este menú contiene un resumen de todas las poses, trayectorias y cuadrículas disponibles del proyecto seleccionado actualmente.

Las poses y las trayectorias se pueden crear, editar, copiar, eliminar e incluso ejecutar directamente. Las cuadrículas, por su parte, son herramientas útiles para mantener organizadas las coordenadas recurrentes y espaciadas uniformemente. Tenga en cuenta que las cuadrículas no se utilizan directamente para mover un robot, sino que agrupan y almacenan conjuntos de valores de coordenadas para mantener su código de aplicación estructurado y evitar que tenga que introducir coordenadas. Se puede extraer una lista de puntos de una cuadrícula para diversos fines; sin embargo, normalmente las cuadrículas se configuran de modo que sus puntos representan directamente coordenadas TCP (Tool Center Point), que luego se utilizan como valores de entrada para las funciones de movimiento.



The screenshot shows a 'Teach-In' window with three sections: 'Available Poses', 'Available Paths', and 'Available Grids'. Each section has a table with columns for Name, Created, and Modified. The 'Available Poses' section shows a folder 'path_1' with sub-items 'end', 'middle', and 'start', and another folder 'path_2' with sub-items 'grid_center' and 'start_pose'. The 'Available Paths' section shows a folder 'path_2' with sub-items 'go_down' and 'go_up', and a folder 'path_1'. The 'Available Grids' section shows three items: 'big_tray', 'pyramid', and 'small_tray'. All items have a 'Created' date of '2019-04-03' and a 'Modified' date of '2019-04-03'.

Available Poses		
Name	Created	Modified
path_1		
end	2019-04-03	2019-04-03
middle	2019-04-03	2019-04-03
start	2019-04-03	2019-04-03
path_2		
grid_center	2019-04-03	2019-04-03
start_pose	2019-04-03	2019-04-03

Available Paths		
Name	Created	Modified
path_2		
go_down	2019-04-03	2019-04-03
go_up	2019-04-03	2019-04-03
path_1	2019-04-03	2019-04-03

Available Grids		
Name	Created	Modified
big_tray	2019-04-03	2019-04-03
pyramid	2019-04-03	2019-04-03
small_tray	2019-04-03	2019-04-03

Figura 4.6: Menú de aprendizaje

	Pulse el botón «Nuevo/Añadir» para abrir un editor de poses, trayectorias o mallas en una nueva ventana. Más adelante en esta sección se explican en detalle todos estos editores.
	Pulse el botón «Filtrar/Buscar» para filtrar la lista de poses, trayectorias o cuadrículas por nombre. Al introducir tres o más caracteres se activa el filtro de búsqueda.
	Pulse el botón «Ejecutar» para mover al robot a una pose o reproducir una ruta. Tenga en cuenta que, al reproducir una ruta, el robot se desplaza primero a la pose inicial de dicha ruta, siempre que no se encuentre ya en esa pose concreta.
	Pulse el botón «Copiar» para crear una copia exacta de esta pose o ruta. El nombre del elemento copiado se modificará ligeramente y terminará en «(copia)».
	Pulse el botón «Editar» (o haga doble clic en cualquier parte del elemento) para abrir esta pose, esta ruta o esta rejilla en una ventana separada para su visualización o edición.
	Pulse el botón «Eliminar» para eliminar de forma irreversible una pose, una ruta o una cuadrícula de su base de datos.
	Haga clic en una carpeta para mostrar u ocultar las poses, las rutas o las cuadrículas que contiene. Los elementos solo se muestran en carpetas (y subcarpetas) si sus nombres contienen uno o varios caracteres «/», por ejemplo, «carpeta/subcarpeta/nombre».

4.6.1 Editor de poses

El editor de poses permite a los usuarios crear una nueva pose o modificar una ya existente. Tenga en cuenta que modificar las coordenadas o la configuración de una pose no mueve automáticamente al robot. Si se desea que esto ocurra, se debe utilizar el control de precisión en su lugar.

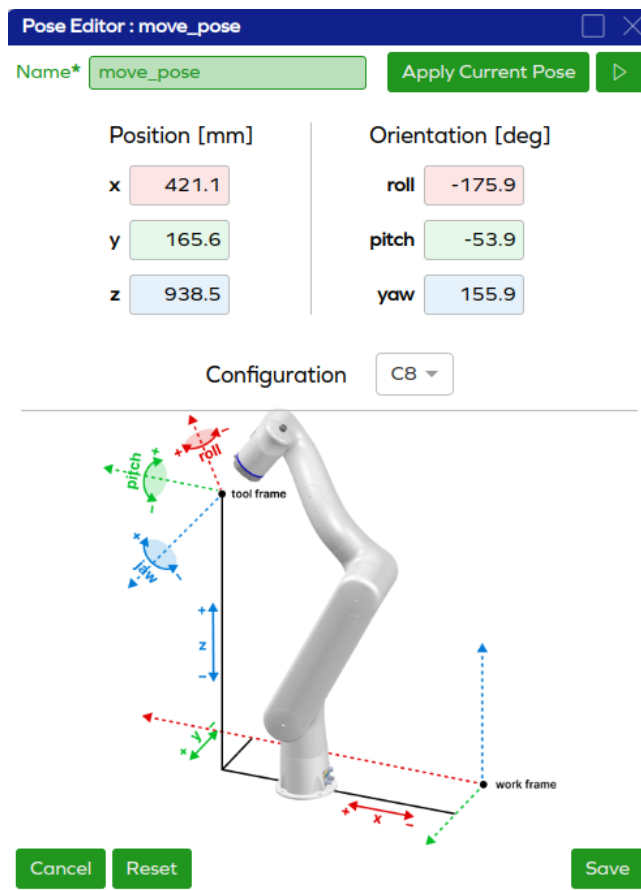


Figura 4.6.1: Editor de poses

Nombre:

Indica el nombre con el que se guardará la pose en la base de datos. Utilice el carácter «/» en el nombre para agrupar sus poses en una estructura similar a una carpeta.

Posición:

Representa la posición del TCP (punto central de la herramienta) en milímetros, definida en relación con el sistema de trabajo global. Tenga en cuenta que se pueden introducir valores de posición no válidos, por ejemplo, aquellos que el brazo robótico no puede alcanzar. Sin embargo, se resaltarán en rojo, ya que pulsar el botón de reproducción provocará un error. Esto está permitido porque la pose puede ser válida si se utiliza con diferentes sistemas de coordenadas de referencia y desplazamientos.

Orientación:

Representa la orientación del TCP en grados, definida con respecto al sistema de trabajo global. La orientación se expresa mediante ángulos náuticos. Al igual que con los valores de posición, los valores de orientación no válidos se resaltan en rojo.

Configuración:

Especifica la configuración del brazo robótico. El usuario puede seleccionar cualquiera de las configuraciones disponibles (C1–C8) mediante el menú desplegable. Tenga en cuenta que cambiar la configuración no modifica la posición ni la orientación del TCP; solo influye en los ángulos de articulación que se utilizan para alcanzar la pose especificada.

Acciones adicionales en el editor de poses

Guardar	Al hacer clic en el botón «Guardar», se guarda la pose actual con el nombre especificado. No se puede guardar una pose sin un nombre válido.
Cancelar	Al hacer clic en el botón «Cancelar», se descartan todos los cambios no guardados que se hayan realizado en el editor y se cierra la ventana.
Restablecer	Al hacer clic en el botón «Restablecer», se deshacen todos los cambios no guardados y la pose vuelve a su estado guardado más recientemente.
 Reproducir	Al hacer clic en el botón «Reproducir» situado en la parte superior de la ventana, se ordena al robot que adopte la pose definida.
Aplicar la pose actual	Al hacer clic en el botón «Aplicar pose actual», se actualizan todos los campos para que reflejen los valores de pose actuales del robot.

4.6.2 Editor de trayectorias

El editor de trayectorias permite a los usuarios crear una nueva trayectoria o modificar una ya existente. Se divide en tres secciones editables: nombre, postura inicial y segmentos de esta trayectoria.

The screenshot shows the 'Path Editor : path_1' window. At the top, there's a 'Name*' field with 'path_1' and a play button. Below it is the 'Initial Pose' dropdown set to 'path_1/initial_pose'. The main section is 'Segments of this path'. It contains two segment cards. The first is a 'Linear' segment with 'Target Pose' set to 'path_1/pose_1', 'Velocity' at 500 mm/s, and 'Acceleration' at 1000 mm/s². Below this is a 'Blend Radius' field set to 100 mm. The second is a 'Circular' segment with 'Intermediate Pose' at 'path_1/pose_2', 'Target Pose' at 'path_1/pose_3', 'Velocity' at 250 mm/s, and 'Acceleration' at 500 mm/s². At the bottom right of the segments list is an 'Add a segment' button with a dropdown set to 'Circular' and a plus icon. At the very bottom are 'Cancel', 'Reset', and 'Save' buttons.

Figura 4.6.2: Editor de trayectorias

Nombre:

Indica el nombre con el que se guardará la ruta en la base de datos. Utilice el carácter «/» en el nombre para agrupar sus rutas en una estructura similar a una carpeta.

Postura inicial:

Define la pose inicial de la ruta. El robot se desplaza primero a esta pose con los ajustes predeterminados de velocidad y aceleración, antes de ejecutar los segmentos definidos uno tras otro.

Segmentos de esta ruta:

Enumera y define los segmentos individuales que componen la ruta. Están disponibles los siguientes tipos de segmentos:

- Segmento lineal
- Segmento circular
- Segmento de coordenadas
- Segmento de posición
- Segmento de orientación

Encontrará una descripción detallada de cada segmento en la sección «Segmento».

Para añadir un nuevo segmento, seleccione el tipo de segmento deseado en el menú desplegable situado en la esquina inferior derecha y haga clic en el botón «+». Una vez añadidos los segmentos, su orden se puede modificar libremente mediante la función de arrastrar y soltar.

Cada segmento requiere la especificación de una pose de destino, que define la pose final de dicho segmento. Las poses de destino deben haberse definido previamente en el editor de poses y pueden seleccionarse a través del menú desplegable. El segmento circular requiere, de forma exclusiva, una pose intermedia adicional, que también debe seleccionarse de la lista de poses previamente definidas.

Para cada segmento se pueden establecer parámetros de velocidad y aceleración. Si no se especifican, se utilizarán los valores globales predeterminados durante la ejecución.

El radio de transición se puede configurar para controlar la suavidad y la eficiencia de las transiciones entre segmentos. Define la desviación máxima permitida en mm con respecto a la pose final del segmento anterior, que corresponde a la pose inicial del segmento actual.

Acciones adicionales en el editor de rutas

El editor de rutas ofrece además funciones adicionales para gestionar, editar y organizar la ruta de forma eficaz.

Guardar	Al hacer clic en el botón «Guardar», la ruta actual se guarda con el nombre especificado. No es posible guardar una ruta sin un nombre válido.
Cancelar	Al hacer clic en el botón «Cancelar», se descartan todos los cambios no guardados que se hayan realizado en el editor y se cierra la ventana.
Restablecer	Al hacer clic en el botón «Restablecer», se deshacen todos los cambios no guardados y la ruta vuelve a su estado guardado más recientemente.
 Reproducir	Al hacer clic en el botón «Reproducir» situado en la parte superior de la ventana, se ordena al robot que ejecute la ruta definida. Si la postura actual del robot no coincide con la postura inicial especificada para la ruta, se desplazará primero a dicha postura inicial antes de que comience la ejecución.
 Eliminar	Al hacer clic en el botón «Eliminar» situado junto a un segmento, este se elimina de la ruta.

4.6.3 Editor de cuadrículas

Al crear una nueva rejilla, se abre el editor de rejillas. Este editor guía al usuario paso a paso por todas las opciones de rejilla disponibles. Utilice los botones «Siguiente» y «Atrás» para navegar por este asistente hasta que la lista ordenada de puntos de rejilla se haya definido según lo deseado. Al editar rejillas ya existentes, el editor de rejillas pasa directamente a la edición manual de puntos.

En un primer paso, el usuario debe seleccionar la dimensión de la rejilla. Por lo general, una rejilla se interpreta como una matriz bidimensional, pero también puede ser una simple lista de puntos sin dependencia de una segunda dimensión (por ejemplo, para seleccionar objetos de una cinta transportadora) o incluso una estructura tridimensional (por ejemplo, para apilar objetos en cajas). La malla se inicia con puntos perfectamente alineados, si bien hay que tener en cuenta que los puntos individuales de la malla pueden modificarse, añadirse o eliminarse en cualquier momento posteriormente.

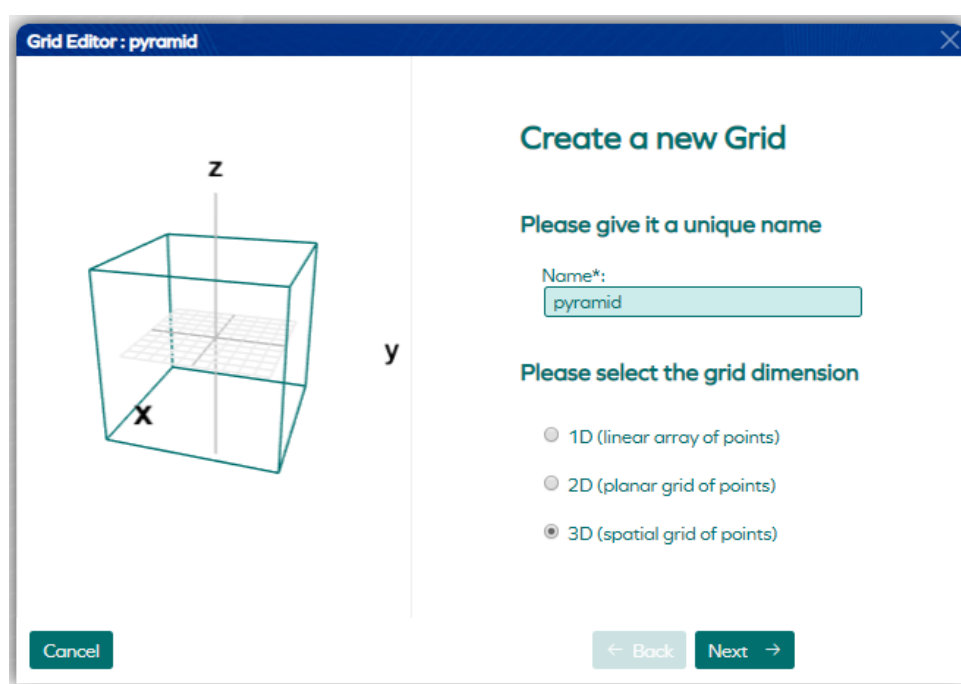


Figura 4.6.3.a: Editor de rejillas (1.º paso)

En un segundo paso, el usuario debe determinar los vértices de la rejilla seleccionada. Las rejillas de diferentes dimensiones tienen un número diferente de vértices, y estos se designan con las letras «A» hasta, por ejemplo, «D» en el caso de una rejilla bidimensional. El usuario puede introducir estos valores de coordenadas (especificados en el sistema de referencia del robot) directamente en los campos de entrada o simplemente guiar el TCP del efector final hacia las esquinas de la rejilla y pulsar los botones «Registrar» correspondientes para leer las coordenadas actuales.

Tan pronto como se hayan programado suficientes vértices para definir la rejilla (por ejemplo, al menos 3 puntos para la rejilla bidimensional), el usuario puede pulsar el botón «Evaluar valores» para evaluar los vértices indicados y completar automáticamente los valores que faltan de tal forma que encajen en un paralelogramo (bidimensional) o en un paralelepípedo (tridimensional). Si además se activa la opción «Alineación automática», el procedimiento tiene en cuenta los ejes y planos de

coordenadas horizontales y verticales e intenta alinear la malla resultante a lo largo de dichos ejes y planos (siempre que estén lo suficientemente cerca).

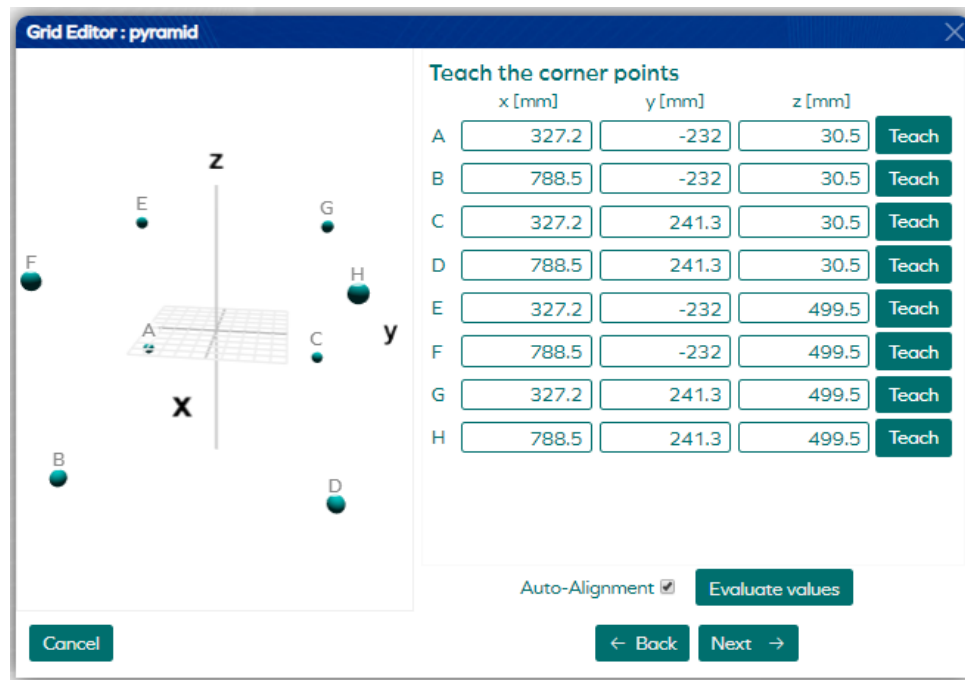


Figura 4.6.3.b: Editor de malla (2.º paso)

En un tercer paso, se solicita al usuario que seleccione el número de puntos, opciones de extensión especiales y direcciones de iteración. Echemos un vistazo a estas opciones disponibles en detalle:

- El número de puntos se puede establecer individualmente para cada dimensión. Si le resulta difícil distinguir las dimensiones (por ejemplo, filas y columnas), consulte las etiquetas de los vértices definidos anteriormente.
- Para las rejillas bidimensionales, existe una opción de extensión que permite añadir puntos intermedios entre cada grupo de cuatro puntos adyacentes. En el caso de las rejillas tridimensionales, esta opción está disponible tanto para cada plano de la rejilla como para el grupo de ocho puntos adyacentes en el espacio.
- La dirección de iteración, es decir, la dirección en la que se devuelven los puntos y se utilizan como iteradores en bucles de programación, se especifica mediante un punto de inicio y un vector en cada dirección. La última dirección viene determinada siempre por las opciones seleccionadas previamente y se muestra únicamente a modo de referencia y para su verificación visual. Las letras de los menús desplegables corresponden a las etiquetas de los vértices de esta malla.

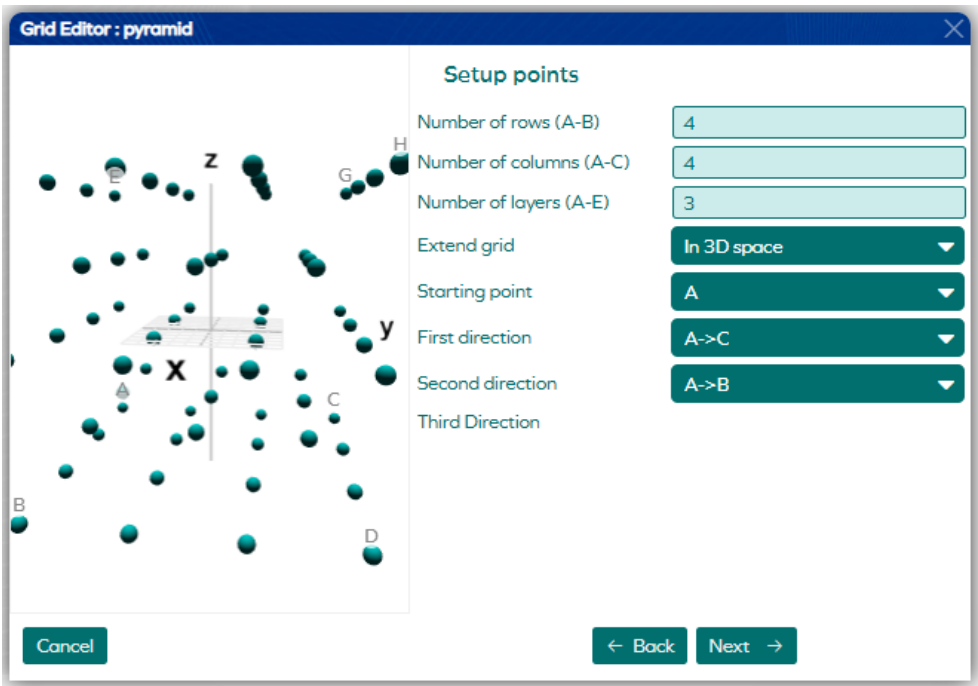


Figura 4.6.3.c: Editor de rejilla (paso 3)

Por último, la lista de puntos de la rejilla queda disponible para su edición manual. De este modo, se pueden copiar, eliminar y mover puntos individuales, así como modificar manualmente los valores de las coordenadas. Al mismo tiempo, la visualización muestra el orden de los puntos y las coordenadas relativas de los puntos de la rejilla.

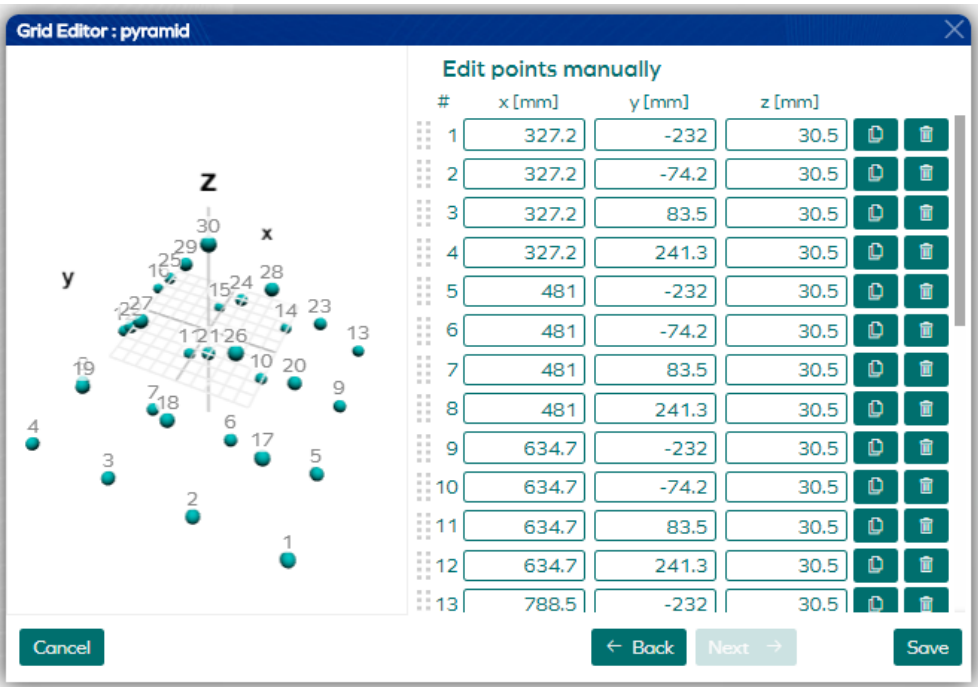


Figura 4.6.3.d: Editor de rejilla (paso 4)

Durante la edición, puede volver atrás en cualquier momento para comprobar las opciones y los valores establecidos anteriormente. También puede modificar estas opciones, aunque al hacerlo se restablecerán todos los cambios realizados en las ventanas siguientes.

4.7 Menú de aplicaciones

Pulse el botón «Aplicaciones» en la ventana principal para acceder al menú de aplicaciones. Este menú contiene un resumen de todas las aplicaciones guardadas, así como la salida de la última aplicación ejecutada. Además, permite al usuario acceder a la ventana «Editor de aplicaciones» para crear nuevas aplicaciones o editar las existentes.

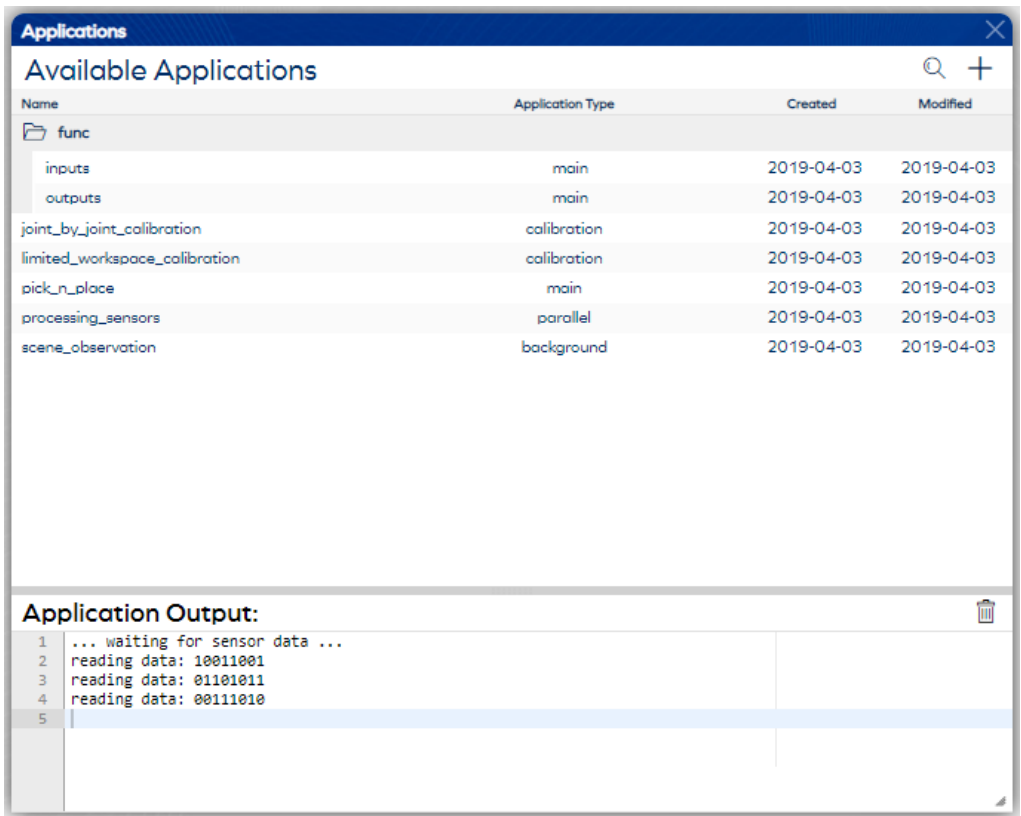


Figura 4.7: Menú de aplicaciones



Pulse el botón «Nuevo/Añadir» para crear una nueva aplicación. De este modo se abrirá el editor de aplicaciones, cuyas funciones se explican detalladamente más adelante en esta sección.



Pulse el botón «Filtrar/Buscar» para filtrar la lista de aplicaciones por nombre. Al introducir tres o más caracteres se activa el filtro de búsqueda.



Pulse el botón «Ejecutar» para ejecutar una aplicación. Tenga en cuenta que de esta forma solo se pueden ejecutar aplicaciones principales, ya que en el Cockpit solo se pueden seleccionar aplicaciones principales.



Pulse el botón «Copiar» para crear una copia exacta de esta aplicación. El nombre del elemento copiado se modifica ligeramente y termina en «(copia)».



Pulse el botón «Editar» (o haga doble clic en cualquier parte del elemento) para abrir esta aplicación en una ventana independiente con el fin de verla o editarla.



Pulse el botón «Eliminar» para eliminar de forma irreversible una aplicación de su base de datos. De este modo se eliminarán tanto el código del programa como la configuración específica de la aplicación.



Haga clic en una carpeta para mostrar u ocultar las aplicaciones que contiene. Los elementos solo se muestran en carpetas (y subcarpetas) si sus nombres contienen uno o varios caracteres «/», por ejemplo, «carpeta/subcarpeta/nombre». Puede hacer clic en el botón «Editar» que aparece para abrir el editor de carpetas.

4.7.1 Editor de aplicaciones

En la sección «Aplicaciones disponibles» del menú de aplicaciones se enumeran todas las aplicaciones guardadas anteriormente. Puede modificar una aplicación existente haciendo clic en el botón «Editar» o crear una nueva aplicación pulsando el botón «Crear nueva aplicación». En ambos casos se abre la ventana «Editor de aplicaciones». Su funcionamiento es similar en ambos casos. Echemos un vistazo rápido a los botones más importantes del editor de aplicaciones antes de entrar en detalles:

Guardar	Al pulsar el botón «Guardar», la aplicación se guarda con el nombre especificado. No es posible guardar una aplicación sin nombre. Utilice el carácter «/» en el nombre para agrupar sus aplicaciones en una estructura similar a la de una carpeta.
Cancelar	Al pulsar el botón «Cancelar», se descartan todos los cambios no guardados que se hayan realizado en este editor y se cierra el editor.
 Probar	Al pulsar el botón «Probar aplicación» situado en la parte superior de la ventana, la aplicación se ejecuta en modo de prueba. El modo de prueba significa que se ejecuta el código del programa, pero no se le impide seguir editando la aplicación. Los cambios añadidos a la aplicación después de probarla no se ejecutan.

4.7.2 Código del programa

La pestaña «Código» de la aplicación es la más importante de las subventanas del editor de aplicaciones. Esta sección muestra el código del programa y, opcionalmente, la biblioteca con funciones y elementos de AX Portal que se pueden insertar en el código del programa. El lenguaje de programación integrado es Python 3.13 con la mayor parte de su funcionalidad básica, incluidos algunos paquetes de Python preinstalados. Además, AX Portal ofrece una amplia biblioteca de funciones que simplifican la escritura de una aplicación robótica (por ejemplo, mover el robot, leer datos de sensores, comunicarse con el usuario, etc.). Encontrará más detalles sobre estas funciones en la documentación del código.

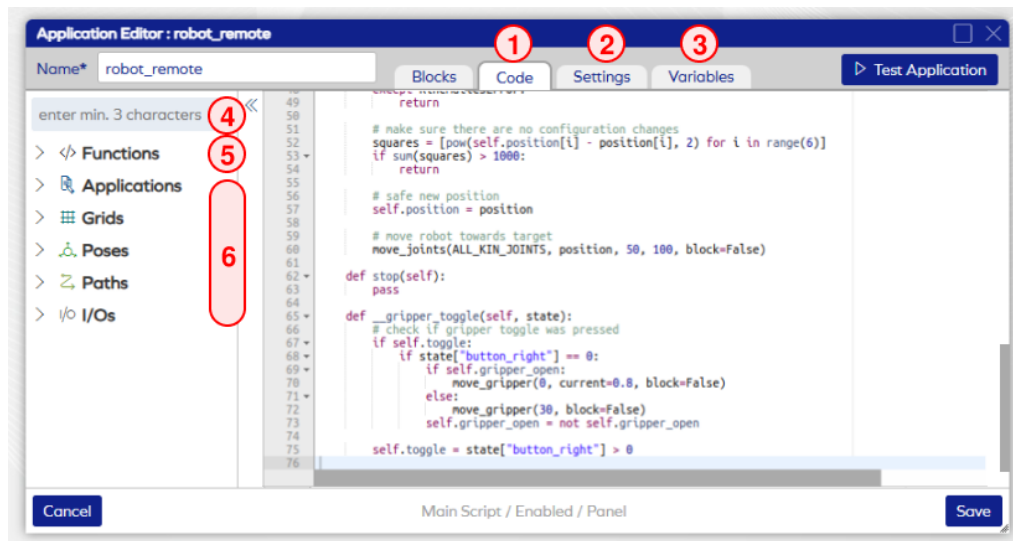


Figura 4.7.2: Editor de aplicaciones (área de código)

1. Escriba aquí su código de programa en Python, 2. Modificar la configuración específica de la aplicación, 3. Definir y modificar variables para la aplicación, 4. Filtro de búsqueda, 5. Biblioteca de funciones, 6. Elementos disponibles para su uso en el programa

Pulse sobre uno de los iconos de funciones o uno de los iconos de elementos situados en la parte izquierda del editor para abrir la paleta de elementos. Aquí encontrará una lista completa de las funciones disponibles. Además, tendrá acceso a todos los elementos que puede insertar en su código con un doble clic, por lo que no es necesario que memorice los nombres de sus elementos.

Consejo: Mejore la legibilidad del código Python eliminando los argumentos opcionales que se insertan al hacer doble clic en una función en su código. Por ejemplo, puede escribir

```
move_joints(joints, joint_position)
```

en lugar de

```
move_joints(joints, joint_position, joint_velocity=None, joint_acceleration=None,  
relative=False, block=True)
```

4.7.3 Configuración de la aplicación

La pestaña «Configuración» contiene algunos ajustes adicionales para la aplicación, entre ellos

- un conmutador «Ejecutable», que puede desactivarse, por ejemplo, si su aplicación aún se encuentra en fase de desarrollo y no debe ejecutarse,
- un conmutador para mostrar esta aplicación en la interfaz del panel,
- una función para cargar y sustituir la imagen que se muestra en la interfaz del panel,
- títulos personalizados como opción para mostrar el nombre de visualización en la interfaz del panel en el idioma de interfaz seleccionado actualmente,
- y, sobre todo, el tipo de aplicación, que se explica en la siguiente sección:

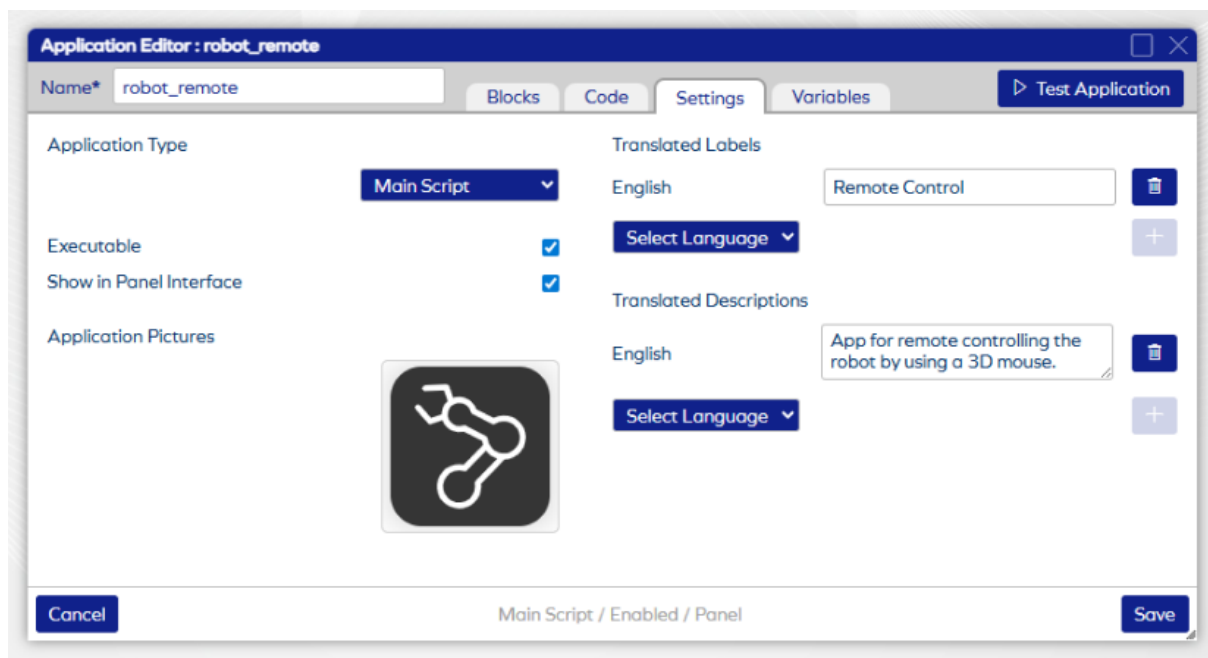


Figura 4.7.3.a: Editor de aplicaciones (área de configuración)

El tipo de aplicación establece en qué contexto debe ejecutarse la aplicación y, por lo tanto, determina cómo puede ejecutarse una aplicación, así como qué funciones pueden utilizarse dentro de ella:

- Las aplicaciones principales son las aplicaciones normales que deben ejecutarse directamente. Están diseñadas para admitir todas las funciones disponibles y controlar directamente el robot y sus dispositivos externos. Por lo general, una aplicación sencilla puede escribirse en un único programa; sin embargo, en el caso de aplicaciones largas o complejas, o para mejorar la legibilidad, puede ser conveniente dividir una aplicación en varios programas que se ejecuten sucesivamente. Sin embargo, varias aplicaciones principales nunca se ejecutan en paralelo; para ello se utilizan las «aplicaciones paralelas». También puede utilizar la instrucción «import» para importar funciones entre aplicaciones principales. Tenga en cuenta que solo las aplicaciones principales pueden seleccionarse a través del panel de control.
- **Las aplicaciones paralelas**, por otro lado, se ejecutan en subprocesos separados y, por lo tanto, están diseñadas para ejecutarse en paralelo a una aplicación principal. Esto puede resultar muy útil, por ejemplo, para la lectura continua de datos de sensores o para aprovechar el tiempo de inactividad del robot para realizar cálculos que requieren un gran esfuerzo computacional. Las aplicaciones paralelas no admiten todas las funciones, como por ejemplo el movimiento del robot, ya que esto complicaría enormemente el flujo del programa o, en caso de falta de atención, podría provocar un comportamiento de movimiento no deseado o incluso peligroso. Todas las aplicaciones paralelas se pausan mientras la aplicación principal está en pausa, y se cierran tan pronto como la aplicación principal se detiene.
- **Las aplicaciones en segundo plano** son aplicaciones que se ejecutan en el fondo de AX Portal independientemente del estado del robot y, por lo tanto, resultan útiles, por ejemplo, para observar, comprobar o registrar datos de forma continua, o para definir variables globales iniciales para un proyecto. Cada proyecto sólo admite una aplicación en segundo plano. Una vez definida, se ejecuta automáticamente al cargar el proyecto correspondiente (por ejemplo, al poner en marcha el robot o al cambiar de proyecto). La aplicación en segundo plano no se ve afectada por las restricciones de estado y, en caso de que contenga un bucle, puede ejecutarse hasta que se apague el robot. Cada vez que se inicia y finaliza una aplicación en segundo plano, se muestra la notificación correspondiente. Puede iniciar, reiniciar o detener la aplicación en segundo plano definida desde el panel de control.



Figura 4.7.3.b: Elementos de control para la aplicación en segundo plano

4.7.4 Variables de la aplicación

La pestaña «Variables» de la aplicación contiene una interfaz de arrastrar y soltar para añadir diferentes tipos de variables de estado a esta aplicación. Las variables de estado solo se pueden utilizar en aplicaciones principales. Se emplean en la interfaz del panel, donde resultan muy útiles, por ejemplo, para supervisar el estado y el progreso de una aplicación o para especificar determinados valores de entrada para una aplicación. Utilice las funciones **set_state_variable(...)** y **get_state_variable(...)** para escribir y leer variables de estado.

A continuación se ofrece una descripción general de los tipos de variables y cómo se representan en la interfaz del panel:

booleano	Los booleanos se representan como simples controles deslizantes que pueden estar activados o desactivados. Se leen y escriben como el tipo bool de Python, donde «activado» corresponde a True y «desactivado» a False .
entero	Los valores enteros se representan como campos de texto que permiten entradas numéricas, incluyendo botones de más y menos para aumentar y disminuir sus valores en el modo de edición. Se leen y escriben como el tipo Python int .
float	Los números de coma flotante se muestran como campos de texto que permiten introducir datos numéricos en el modo de edición. Se leen y se escriben como el tipo float de Python.
slider	Los números de coma flotante se representan en el modo de edición como controles deslizantes con un valor mínimo y máximo obligatorio. Se leen y escriben como el tipo float de Python .
texto	Los textos se muestran en el modo de edición como campos de texto. Se leen y escriben como el tipo str de Python .
dropdown	Las variables con una selección específica se muestran en el modo de edición como listas desplegables. Se leen y escriben como el tipo str de Python .

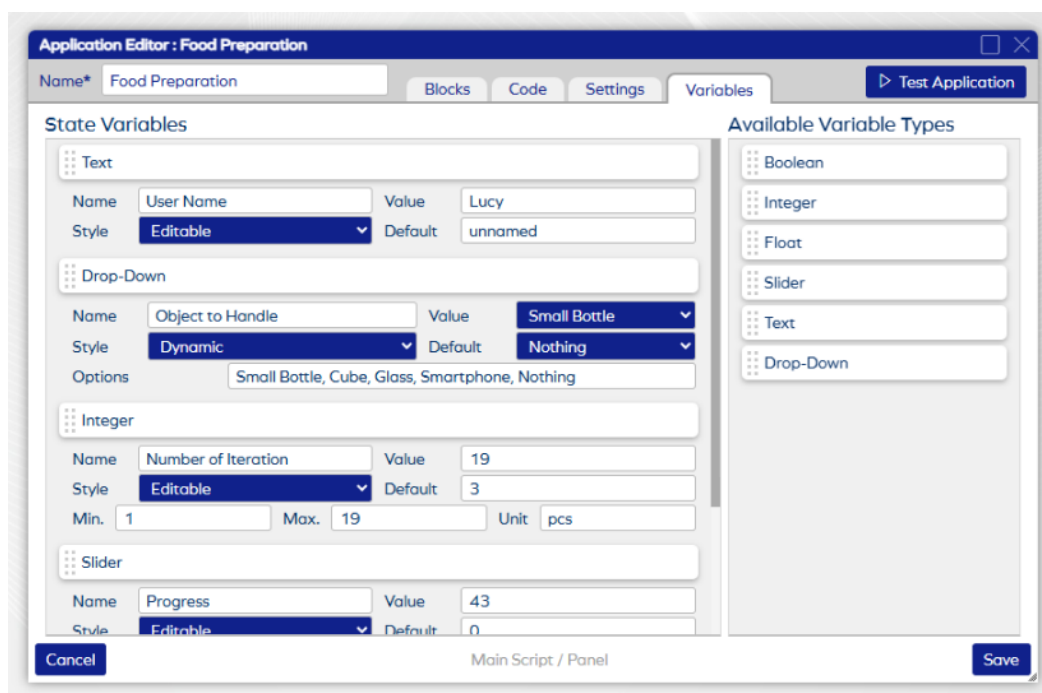


Figura 4.7.4: Editor de aplicaciones (área de variables)

A continuación se ofrece una descripción general de las opciones, que varían en función del tipo de variable:

Nombre	Asigne a su variable un nombre adecuado; posteriormente podrá acceder a ella con ese nombre en forma de cadena de caracteres.
Valor	El valor de una variable contiene el valor actualmente asignado a dicha variable, que se lee y se escribe en el tipo de Python correspondiente, tal y como se ha explicado anteriormente.
Predeterminado	El valor por defecto de una variable es el valor que adopta cuando las variables se restablecen mediante el botón correspondiente en la interfaz del panel.
Estilo	El estilo de una variable define si, cuándo y cómo se puede editar dicha variable.
Mín.	Las variables numéricas permiten establecer un límite inferior para dicho número. Si introduce valores por debajo del mínimo, se aplicará el valor mínimo. Los límites son obligatorios para los controles deslizantes y opcionales para otras variables.
Máx.	Las variables numéricas permiten establecer un límite superior para este número. Si introduce valores por encima del máximo, se aplicará el valor máximo. Los límites son obligatorios para los controles deslizantes y opcionales para otras variables.
Unidad	En el caso de las variables numéricas, puede añadir opcionalmente una unidad como cadena de caracteres al número.
Opciones	Las opciones para las variables desplegables pueden introducirse como una cadena de caracteres separada por comas.

A continuación se ofrece una descripción general de los estilos de variable, que determinan cómo se utiliza una variable:

- **Las variables editables** permiten modificar el valor antes de ejecutar la aplicación, pero no durante la ejecución.
- **Las variables protegidas** son editables, pero es necesario pulsar un botón de edición adicional para cambiar el valor y un botón de guardar adicional para aplicar el cambio de valor.
- **Las variables dinámicas** son editables y sus valores pueden modificarse dinámicamente incluso en tiempo de ejecución.
- **Las variables dinámicas protegidas** comparten las características de las variables protegidas y dinámicas y, por lo tanto, se pueden editar en tiempo de ejecución pulsando una tecla de edición.
- **Las variables de solo lectura** no son editables; su valor solo puede ser modificado por la aplicación. Se utilizan habitualmente para supervisar el estado o el progreso de una aplicación.
- **Las variables ocultas** no se muestran en la interfaz del panel, pero se pueden escribir y leer de la misma manera que otras variables de estado.

4.7.5 Variables compartidas y globales

Además de las variables de estado específicas de la aplicación explicadas anteriormente, se pueden definir variables compartidas y globales para compartir datos entre varios scripts que se ejecutan en paralelo o entre varias sesiones de la misma aplicación:

- **Las variables compartidas** solo existen durante el tiempo de ejecución de una única aplicación. Utilice las funciones **set_shared_variable(...)** y **get_shared_variable(...)** para compartir valores entre varias aplicaciones principales y/o aplicaciones en paralelo.
- **Las variables globales** no solo comparten valores entre aplicaciones en ejecución, sino que también persisten tras reiniciar el robot. Para ello, utilice las funciones **set_global_variable(...)** y **get_global_variable(...)**.

4.7.6 Editor de carpetas

Si el nombre de la aplicación contiene el carácter «/», la aplicación se clasifica automáticamente en una estructura de carpetas. Puede editar una carpeta haciendo clic en el botón de edición que aparece al pasar el cursor por encima de dicha carpeta. De este modo se abre el editor de carpetas. Este editor ofrece la posibilidad de mostrar el nombre de la carpeta en el idioma de interfaz seleccionado (de forma similar al editor de aplicaciones). Los títulos de las carpetas, así como los títulos de las aplicaciones, aparecen en la interfaz del panel; de este modo, la interfaz del panel puede configurarse fácilmente para que sea comprensible en varios idiomas.

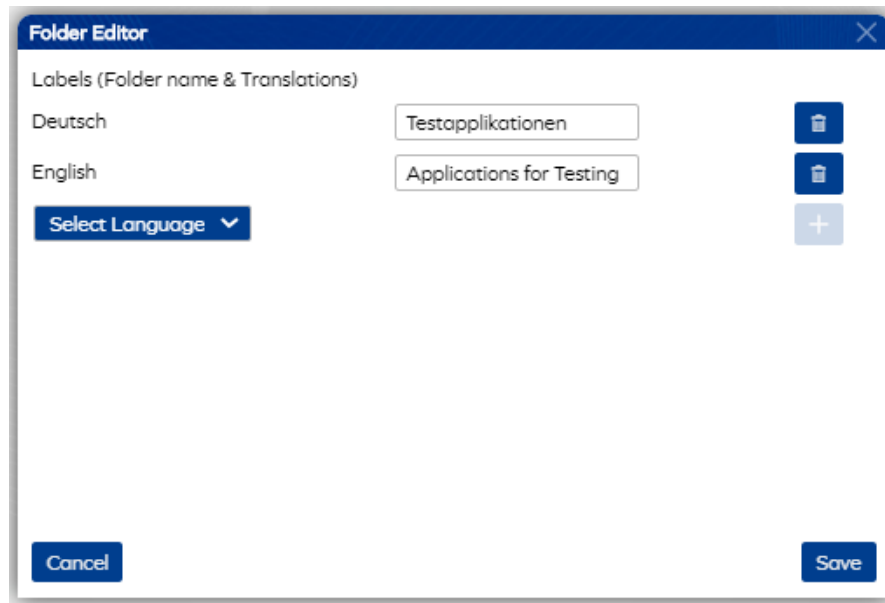


Figura 4.7.6: Editor de carpetas

4.7.7 Programación por bloques

La pestaña «Bloques» del editor de aplicaciones contiene una interfaz de programación gráfica. Esta interfaz permite al usuario crear secuencias de programa sencillas arrastrando y soltando bloques de programación en el área de programa.

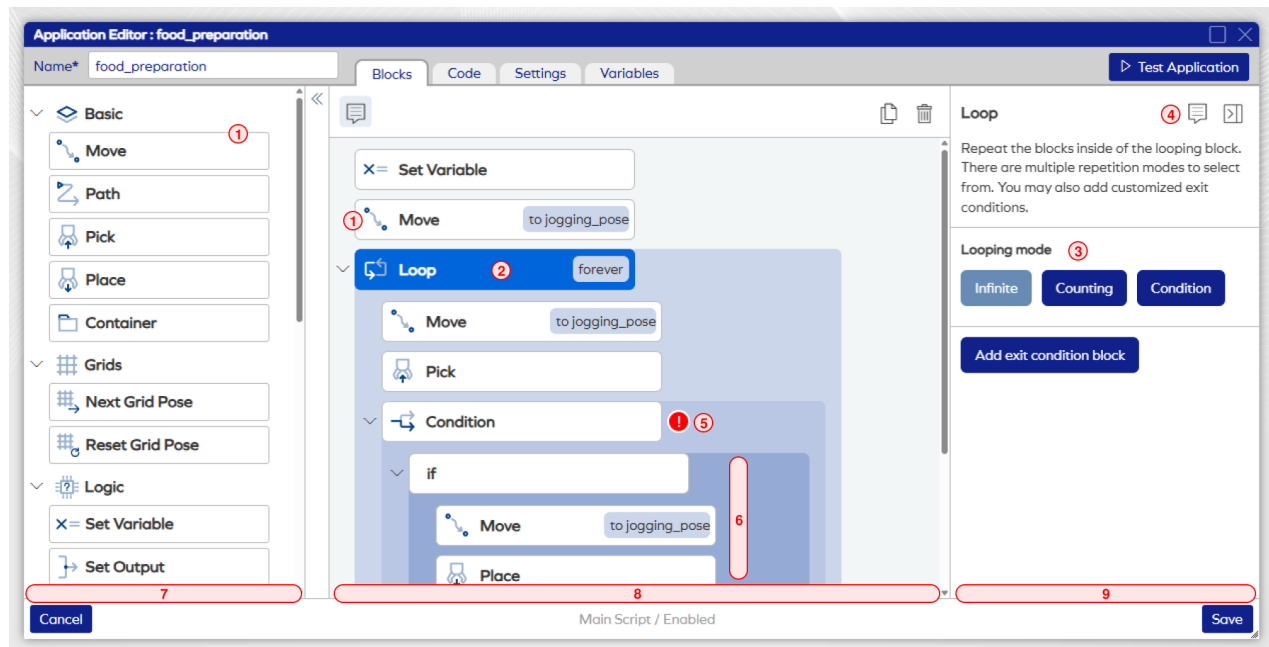


Figura 4.7.7: Editor de aplicaciones (programación por bloques)

1. Arrastrar bloques al área de programa, 2. Seleccionar bloque para editar, 3. Modificar parámetros,
4. Comentario, 5. Indicador de errores, 6. Contenedor de bloques, 7. Bloques disponibles, 8. Área de programa, 9. Detalles del bloque seleccionado

Seleccione un bloque en el flujo del programa para ver sus detalles. Algunos parámetros son obligatorios para que el programa sea ejecutable (por ejemplo, en un bloque de movimiento debe proporcionar información sobre hacia dónde debe desplazarse el robot). Otros parámetros, por su parte, son opcionales (por ejemplo, se utilizan valores predeterminados si no se especifica nada) o pueden utilizarse para mejorar la legibilidad del programa (por ejemplo, comentarios). Si un bloque no está definido correctamente (por ejemplo, falta un parámetro obligatorio o se ha especificado un valor no válido), mostrará un mensaje de error en el flujo del programa y, al seleccionarlo, se mostrará el motivo del error en los detalles.

Algunos bloques, al añadirlos, crean una estructura anidada resaltada en color en el flujo del programa. Se trata de una ayuda gráfica para indicar que estos bloques están agrupados y se ejecutan siempre en el orden (de arriba abajo). Los bucles y las condiciones son ejemplos de este tipo de bloques, que se encargan, por ejemplo, de ejecutar un grupo de bloques varias veces o de ejecutar un grupo solo si se cumple una condición determinada.

Además, un bloque seleccionado puede eliminarse o copiarse (y arrastrarse a la posición deseada en el programa).

4.7.7.1 Uso de variables de aplicación y expresiones

Antes de explicar en detalle todos los bloques de programación, tenga en cuenta que la mayoría de los campos de texto de los parámetros de los bloques admiten expresiones matemáticas y/o lógicas. Estos campos también admiten la inserción de valores de variables de aplicación predefinidas, así como de valores de entradas digitales y analógicas.

La forma más sencilla de explicarlo es mediante un ejemplo. Supongamos que hemos definido las siguientes variables:

The screenshot shows a 'State Variables' configuration window. It contains two sections: 'Boolean' and 'Integer'.
 In the 'Boolean' section, there is a variable named 'flag' with a style set to 'Editable'.
 In the 'Integer' section, there is a variable named 'counter' with a style set to 'Editable', a minimum value of 0, a maximum value of 1000, and a default value of 0.

Figura 4.7.7.1.a: Variables de ejemplo

Ejemplos de **expresiones numéricas** válidas:

Four examples of valid numeric expressions are shown in a text field:
 1. 42
 2. $-7 * (8.2 - 3)$
 3. $\{counter\} + 1$
 4. $\{Tool AIO 02\}$

Ejemplos de **expresiones lógicas** válidas:

Four examples of valid logical expressions are shown in a text field:
 1. FALSE
 2. NOT {flag}
 3. {flag} AND ({counter} > 5)
 4. {Main DIO 09}

Figura 4.7.7.1.b: Ejemplo de expresiones numéricas y lógicas

Los valores de las variables y las entradas también se pueden introducir en campos de texto normales:

An example of a text expression is shown in a text field: "The counter is set to {counter}."

Figura 4.7.7.1.c: Ejemplo de una expresión de texto

4.7.7.2 Bloque de movimiento

El **bloque de movimiento** ejecuta una función de movimiento que desplaza al robot desde su posición actual hasta un destino. El usuario debe seleccionar el estilo de movimiento...

- **Punto a punto** → Desplazamiento hacia el destino de la forma más rápida posible, sin restricciones en la trayectoria
- **Lineal** → Desplazamiento hacia el destino a lo largo de una trayectoria lineal; la orientación se interpola linealmente

- **Circular** → Desplazamiento hacia el destino a lo largo de una trayectoria circular; la orientación se interpola linealmente, y el usuario debe especificar **posición intermedia** en la trayectoria circular

... y el formato de destino.

- **Nombre de la pose** → Seleccione el nombre de la pose de destino (guardada previamente)
- **Nombre de la rejilla** → Desplazarse a los puntos de una rejilla (guardada previamente); primero se desplaza al primer punto de esta rejilla a continuación se puede utilizar el **bloque «Pose de rejilla siguiente»** para incrementar el contador de puntos de la rejilla
- **Ángulo de articulación** → Especifique los ángulos de articulación de la pose de destino
- **Coordenadas** → Indique las coordenadas de la pose de destino

Además, el usuario puede modificar los siguientes ajustes opcionales. Tenga en cuenta que, si no se especifican, se utilizarán los valores predeterminados/globales.

- **Ajustes de velocidad**
 - **Velocidad/aceleración de la articulación** → Indique la velocidad para los movimientos de punto a punto
 - **Velocidad/aceleración lineal** → Indique la velocidad para movimientos lineales/circulares
 - **Velocidad/aceleración de rotación** → Indique la velocidad para los movimientos que solo cambian la orientación del TCP (la posición permanece igual)
- **Ajustes del sistema de coordenadas de referencia**
 - **Desplazamiento de la herramienta** → Desplaza el TCP con respecto al sistema de brida (parte superior del brazo del robot)
 - **Sistema de coordenadas de la herramienta** → Modifica el TCP con respecto al sistema de brida (parte superior del brazo robótico)
 - **Desplazamiento de trabajo** → Desplaza el sistema de coordenadas de referencia respecto al sistema base (parte inferior del brazo robótico)
 - **Sistema de coordenadas de trabajo** → Modifica el sistema de coordenadas de referencia con respecto al sistema base (parte inferior del brazo robótico)
- **Otros ajustes**
 - **Coordenadas relativas** → Si se activa esta opción, los valores de las coordenadas se suman a las coordenadas actuales
 - **Flujo de programa bloqueado** → Si se activa esta opción, el flujo del programa no continúa hasta que se haya completado el movimiento

4.7.7.3 Bloque de trayectoria

El **bloque de** trayectoria reproduce una trayectoria definida previamente. El usuario selecciona la trayectoria que se va a ejecutar. Al ejecutar este bloque, el robot se desplaza primero a la posición inicial de la trayectoria y, a continuación, recorre los segmentos de la trayectoria en un movimiento continuo.

4.7.7.4 Bloque de recogida

El **bloque «Recoger»** envía una orden de recogida a la herramienta conectada. En el caso de pinzas angulares o paralelas, esto significa cerrar la pinza para mantener el objeto en su interior. En el caso

de pinzas neumáticas, esto significa activar la bomba de vacío para mantener el objeto en su sitio. Este bloque no tiene parámetros.

4.7.7.5 Bloque «Colocar»

El **bloque «Colocar»** envía un comando de colocación a la herramienta conectada. En el caso de pinzas angulares o paralelas, esto significa abrir la pinza para liberar el objeto que se encuentra en su interior. En el caso de pinzas neumáticas, esto significa desactivar la bomba de vacío para depositar el objeto en su lugar. Este bloque no tiene parámetros.

4.7.7.6 Bloque «Contenedor»

El **bloque Contenedor** ofrece una forma sencilla de agrupar varios bloques. No tiene asignada ninguna función especial.

4.7.7.7 Bloque «Siguiente pose de rejilla»

El **bloque «Pose de rejilla siguiente»** incrementa el contador de pose interno para una rejilla determinada. Debe seleccionarse la rejilla en cuestión. Siempre que se utiliza un **bloque de movimiento** para desplazarse a una rejilla, se apunta a la pose a la que apunta actualmente este contador. Para desplazarse efectivamente a todos los puntos de una rejilla, puede utilizar tanto el **bloque de movimiento** como el **bloque de pose de rejilla siguiente** dentro de una estructura de bucle de la siguiente manera:

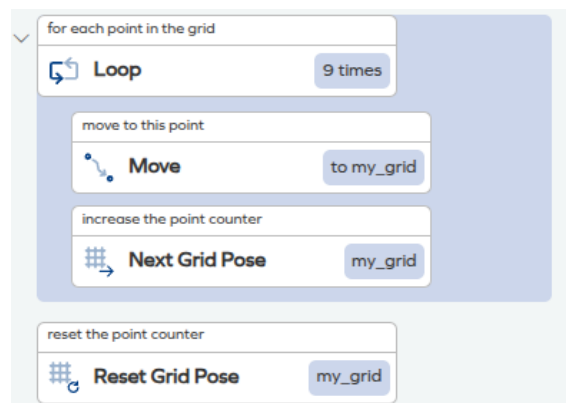


Figura 4.7.7.7: Ejemplo de uso de bloques de rejilla

4.7.7.8 Bloque «Restablecer pose de rejilla»

El **bloque «Restablecer pose de rejilla»** restablece a cero (valor inicial) el contador de poses interno de una rejilla determinada. Debe seleccionarse la rejilla en cuestión.

4.7.7.9 Bloque «Escribir variable»

El **bloque «Escribir variable»** establece el valor de una variable de aplicación predefinida. Es imprescindible seleccionar primero una variable y, a continuación, un valor. Se permite el uso de expresiones numéricas y/o lógicas.

4.7.7.10 Bloque «Establecer salida»

El **bloque** «Establecer salida» establece el valor de una salida digital o analógica. El usuario debe seleccionar primero la salida que desea configurar; dependiendo de si se trata de una salida digital o analógica, el valor que se debe establecer es un valor booleano o un número.

4.7.7.11 Bloque Esperar

El **bloque** «Esperar» bloquea la ejecución del programa hasta que se produzca un evento determinado. Se puede elegir entre tres modos diferentes:

- **Tiempo** → Esperar hasta que haya transcurrido un intervalo de tiempo determinado (se permite la expresión numérica)
- **Movimiento** → Esperar hasta que se detenga un comando de movimiento en curso
- **Entrada** → Esperar hasta que el valor de una entrada digital se establezca en un valor determinado o hasta que el valor de una entrada analógica supere un umbral determinado

4.7.7.12 Bloque «Bucle»

El **bloque de bucle** crea un grupo de bloques que se ejecuta de forma repetida. Hay tres modos diferentes entre los que elegir:

- **Infinito** → Bucle que se ejecuta hasta que se interrumpe la aplicación
- **Contador** → Bucle para un número determinado de repeticiones (se permite la expresión numérica)
- **Condición** → Bucle hasta que se cumpla una condición determinada (se permiten expresiones lógicas)

El usuario puede añadir un **bloque** «Salir si» al bucle; consulte la siguiente sección.

4.7.7.13 Bloque «Salir si»

El **bloque** «Salir si» está vinculado a su **bloque de bucle** y sale de este bucle cuando se cumple la condición especificada (se permiten expresiones lógicas).

4.7.7.14 Bloque de condición

El **bloque de condición** crea un contenedor con dos grupos, «if» y «else», en su interior. La condición (se permite el uso de expresiones lógicas) se utiliza para derivar el flujo del programa hacia el grupo «if» (si se cumple la condición) o hacia el grupo «else» (si no se cumple la condición).

4.7.7.15 Bloque de comentario

El **bloque de comentario** simplemente inserta un comentario informativo en el flujo del programa. Tenga en cuenta que cualquier otro bloque también ofrece la posibilidad de añadir un comentario, que luego será visible en el área del programa. Sin embargo, en ocasiones resulta útil crear un comentario independiente que no esté vinculado a un bloque concreto.

4.7.7.16 Bloque «Finalizar»

El **bloque** «Finalizar» interrumpe el programa que se está ejecutando actualmente. Los bloques de programación restantes ya no se ejecutarán.

4.7.7.17 Bloque de diálogo

El **bloque de diálogo** crea un contenedor con varios grupos en su interior. Durante la ejecución, se interrumpe el flujo del programa y aparece un cuadro de diálogo. El usuario debe responder al diálogo para que el flujo del programa continúe. Los grupos del interior (por defecto «Aceptar» y «Cancelar») corresponden a los botones del cuadro de diálogo. Dependiendo del botón pulsado, el flujo del programa se ramifica y continúa con el grupo seleccionado. El mensaje del cuadro de diálogo (se permite la inyección de variables), los nombres de los botones y el número de botones se pueden modificar en los parámetros.

4.7.7.18 Bloque de salida

El **bloque de salida** muestra un mensaje definido por el usuario en la salida de texto.

4.7.7.19 Bloque de registro

El **bloque de registro** registra un mensaje definido por el usuario en un archivo de registro definido por el usuario. Los archivos de registro se pueden descargar a través de la interfaz de escritorio, por ejemplo, con la combinación de teclas CTRL + MAYÚS + L.

4.7.7.20 Bloque de notificación

El **bloque de notificación** envía una notificación definida por el usuario a la interfaz gráfica de usuario. La notificación no bloquea el flujo del programa.

4.7.7.21 Bloque de advertencia

El **bloque de advertencia** envía un mensaje de advertencia definido por el usuario a la interfaz gráfica de usuario (GUI). La advertencia no bloquea el flujo del programa.

4.7.7.22 Bloque de error

El **bloque de error** envía un mensaje de error definido por el usuario a la GUI e interrumpe el programa que se está ejecutando actualmente.

4.7.7.23 Bloque de código Python

El **bloque de código Python** es un bloque avanzado que permite al usuario ejecutar unas cuantas líneas de código Python independiente. Puede utilizar este bloque para llamar a funciones que, de otro modo, no estarían disponibles mediante la programación por bloques. En su interior se pueden utilizar todas las funciones.

4.8 Menú E/S

Pulse el botón «E/S» en la ventana principal para acceder al menú de E/S. Este menú muestra todas las entradas y salidas digitales y analógicas, así como sus propiedades, que se han definido previamente en la ventana de configuración de E/S (su configuración, funciones asignadas y nombres definidos por el usuario), además de los valores de todas las entradas y salidas. Las entradas son de solo lectura; por lo tanto, estos valores pueden supervisarse fácilmente aquí. Las salidas, por el contrario, también pueden configurarse a través de esta ventana.

Las E/S digitales representan valores booleanos, donde «True» o «1» o «verde» significa «alimentado» y «False» o «0» o «gris» significa «sin alimentación». Las entradas digitales

muestran este valor en un indicador circular, mientras que las salidas digitales pueden modificarse pulsando el botón de visualización cuadrado para cambiar su estado.

Las E/S analógicas se representan como **valores de coma flotante** en incrementos de 0,1. La unidad de una E/S analógica se determina mediante la configuración de dicha E/S (V para tensión, mA para corriente). Mientras que las entradas analógicas se muestran como un número observable, las salidas analógicas pueden modificarse mediante los campos de texto situados junto al valor actual de dicha salida.

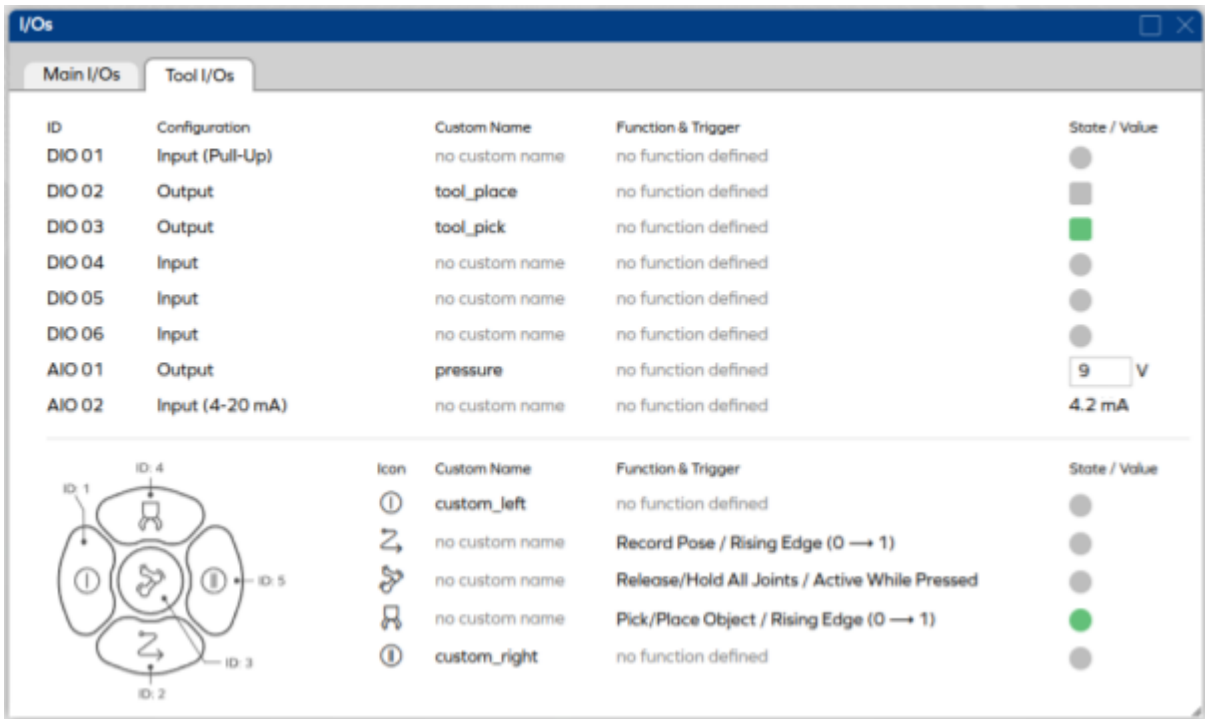


Figura 4.8: Menú de E/S

4.9 Menú de visualización

Pulse el botón «Visualización» en la ventana principal para acceder al menú de visualización. Esta ventana muestra la posición y la orientación actuales de las articulaciones, lo que resulta útil, por ejemplo, cuando se maneja el robot a distancia sin retroalimentación visual directa. También puede ver las poses guardadas anteriormente en la visualización 3D. La visualización es interactiva y permite realizar algunas acciones de control:

- Gire la imagen haciendo clic con el ratón dentro de la visualización (botón izquierdo del ratón) y arrastrando la imagen hacia los lados, así como hacia arriba y hacia abajo.
- Acérquese y aléjese girando la rueda del ratón hacia arriba o hacia abajo.
- Desplace la cámara lateralmente, así como hacia arriba y hacia abajo, utilizando las teclas de flecha de su teclado, o haga clic con el botón derecho del ratón y arrastre hacia la izquierda y hacia la derecha.

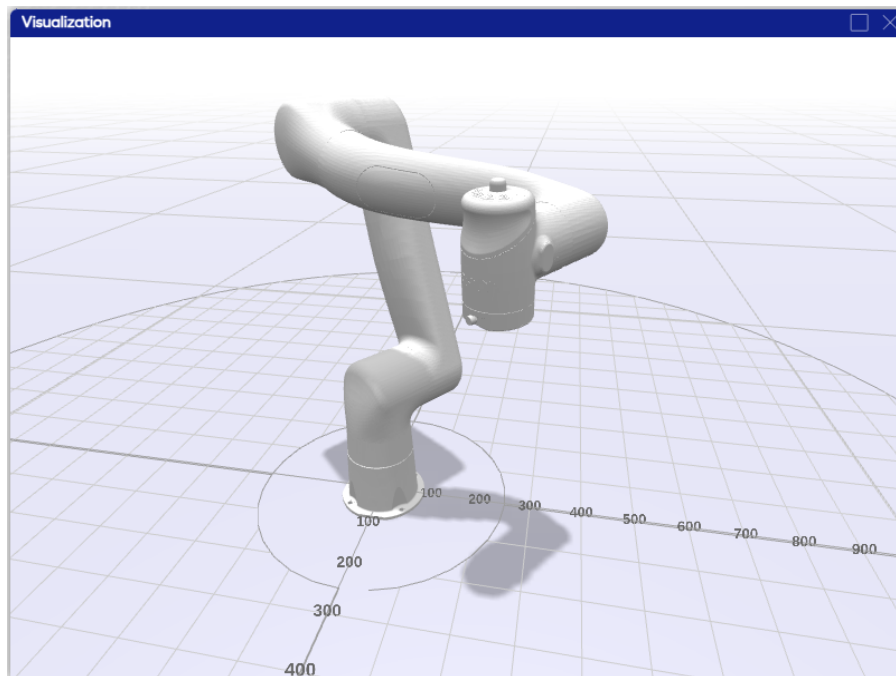


Figura 4.9: Ventana de visualización

4.10 Control manual

Pulse el botón «Control manual» en la ventana principal para abrir el menú de control manual. En este menú, el usuario puede especificar qué articulaciones del robot pueden moverse durante el control manual. De forma predeterminada, al pulsar el botón HGC del robot (y el botón correspondiente en la cabina), el usuario puede mover todas las articulaciones. Al hacer clic en los indicadores de cada articulación de este menú, el usuario puede activar o desactivar articulaciones individuales para el control manual.

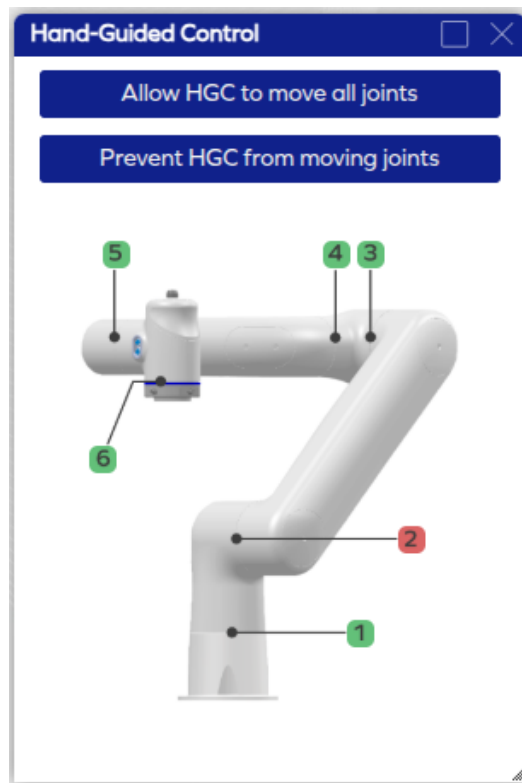


Figura 4.10: Menú para el funcionamiento manual

4.11 Monitor del sistema

Pulse el botón «Monitor del sistema» en la ventana principal para abrir el monitor del sistema. Este menú contiene los siguientes datos del sistema que el usuario puede supervisar:

4.11.1 Estado del robot

La pestaña «Estado del robot» muestra el estado actual del robot. Esto incluye las coordenadas del punto central de la herramienta (TCP) en relación con el sistema base del robot, así como las posiciones, corrientes y valores de temperatura de todos los motores. Además, aquí se pueden ver los estados de todas las entradas de seguridad.

4.11.2 Salida de texto

Tan pronto como una aplicación devuelve texto mediante la función **print(...)**, dicho texto aparece en la pestaña «Salida de texto» de esta ventana. Tenga en cuenta que, en el caso de aplicaciones que se ejecutan en paralelo, todas las aplicaciones pueden escribir texto en la salida de texto.

4.11.3 Registro de aplicaciones

Tan pronto como una aplicación registra datos mediante la función `log_message(...)`, dichos datos aparecen en la pestaña «Registro de la aplicación» de esta ventana. Aquí se pueden filtrar y buscar los datos registrados. Estos registros definidos por el usuario se descargan junto con los archivos de registro integrados al utilizar la combinación de teclas CTRL + MAYÚS + L.

4.11.4 Historial de mensajes

Tan pronto como se muestra un mensaje en la interfaz de usuario, este también aparece en la pestaña «Historial de mensajes» de esta ventana. Los mensajes se dividen en cuatro tipos, que se utilizan en diferentes contextos:



Los mensajes informativos proporcionan al usuario indicaciones útiles, comunicaciones informativas o consejos.



Los mensajes de éxito informan al usuario de que un comando o una acción se ha ejecutado correctamente.



Las advertencias proporcionan al usuario información importante que debe leerse con atención antes de continuar. Las advertencias nunca son críticas y no interrumpen el flujo del programa.



Existe una amplia gama de **errores**, desde pequeños errores de manejo (por ejemplo, acciones incorrectas o argumentos de función erróneos) hasta errores críticos, causados, por ejemplo, por un robot dañado.

4.12 Menú de ayuda

El menú de ayuda contiene información útil sobre el sistema actualmente en ejecución y los módulos instalados, sobre los atajos de teclado a disposición del usuario, así como sobre las opciones de contacto en caso de preguntas o problemas.

4.12.1 Información del robot

La pestaña «Información del robot» del menú de ayuda contiene información detallada sobre el entorno AX Portal y los componentes de hardware conectados (si los hay). En la sección «Módulos» de la información del robot encontrará una lista de los módulos instalados, así como información sobre si se han cargado correctamente. Solo si se muestra el estado «OK» podrá utilizar las funciones y los menús que ofrece este módulo. En las solicitudes de asistencia, indique siempre la información detallada del robot de AX Portal y del hardware conectado. Simplemente pulse el botón «Copiar al portapapeles» y pegue el contenido (a ser posible en una [fuente de ancho fijo](#)) junto con su solicitud de asistencia.

4.12.2 Atajos de teclado

La pestaña «Atajos de teclado» contiene un resumen de las combinaciones de teclas disponibles. Las combinaciones de teclas se dividen en combinaciones generales, que se utilizan, por ejemplo, para manejar los botones de control del panel de control, y atajos de teclado para el editor de aplicaciones, que resultan útiles para programar aplicaciones de AX Portal. Tenga en cuenta que la distribución del teclado en macOS difiere considerablemente de la de otros sistemas operativos (Windows, Linux) y, por lo tanto, la lista de combinaciones de teclas también es diferente.

4.12.3 Documentación del código

La pestaña «Documentación del código» contiene la documentación de todas las funciones disponibles para la programación de las aplicaciones. La documentación se divide en funciones básicas, que siempre están disponibles, y funciones adicionales, que se refieren a módulos instalados adicionalmente. Mediante el botón «Descargar» situado en la parte superior de la pestaña, se puede descargar la documentación del código como archivo ZIP. Para consultar la documentación del código sin conexión, abra el archivo «index.html» en su navegador.

4.12.4 Licencia de código abierto

La pestaña «Licencia de código abierto» contiene toda la información necesaria sobre las bibliotecas y módulos de código abierto utilizados por AX Portal.

4.12.5 Información de contacto

La pestaña «Contacto» muestra la información de contacto del servicio de asistencia. Consulte siempre primero los manuales antes de enviar solicitudes de asistencia. Cuando envíe solicitudes de asistencia, asegúrese de descargar los archivos de registro y de adjuntarlos a su solicitud.

4.13 Menú de configuración

En el menú de configuración, el usuario puede modificar de forma permanente algunos de los ajustes básicos de AX Portal. Pulse «Guardar» para aplicar todos los cambios, «Cancelar» para descartarlos o «Restablecer» para restablecer todos los ajustes a los valores guardados por última vez. A continuación se explican todas las categorías y opciones:

The screenshot displays the 'Settings' window of the AX Portal, organized into several sections:

- GUI Settings:**
 - Live Auto-Completion: ☒
 - Show Optional Function Arguments: ☒
 - Message Fading Level: Warning (dropdown)
 - Message Fading Time: 5 (slider)
- Customize Cockpit Buttons:**
 - Show 'Hand-Guided Control' button: ☒
 - Show 'Gripper' button: ☒
 - Show 'Jogging' button: ☒
 - Show 'Motion Controls' button: ☒
 - Show 'Homing' button: ☒
 - Homing Position:
 - Joint 1: 0
 - Joint 2: 0
 - Joint 3: 0
 - Joint 4: 0
 - Joint 5: 0
 - Joint 6: 0
- Start-Up Settings:**
 - Start-Up Project: Last Selected (dropdown)
 - Start-Up Application: Last Selected (dropdown)
 - Triggering Events: No Trigger (dropdown)
- Motion Behaviors:**
 - Interrupt Behavior: Pause and Approve (dropdown)
 - Collision Resume Attempts: 3 (slider)
 - Collision Resume Timeout: 20 (slider)
 - Temperature Speed Reduction: ☒
- Authentication Settings:**
 - Disable Authentication for ROS: ☒
 - Disable Authentication for TCP: ☒
 - Disable Authentication for REST: ☒
 - Disable Authentication for Modbus: ☒
- Default TCP Communication Settings:**
 - Encoding Standard: UTF-8
 - End Character: \x03
 - Send Heartbeat: ☒
- ROS Function Settings:**
 - Discovery Mode: Peer to Peer (dropdown)
 - IP Addresses of ROS 2 Peers: 127.0.0.1 (text input with add/remove buttons)
 - ROS Domain ID: 10 (dropdown)

At the bottom, there are buttons for 'Cancel', 'Reset', and 'Save'.

Figura 4.13: Menú de configuración

4.13.1 Ajustes de la interfaz gráfica de usuario

Esta categoría contiene varios ajustes relacionados con la interfaz de usuario.

Autocompletado de texto :	Active esta opción para ver sugerencias de autocompletado de texto en tiempo real mientras escribe código de programa. Si está desactivada, aún así puede completar automáticamente la palabra actual mediante el atajo de teclado CTRL+ESPACIO.
Mostrar argumentos opcionales	Seleccione si desea que se incluyan o no los argumentos opcionales de las funciones al insertarlas en el código del programa. Para mantener el código libre de argumentos de función innecesarios, le recomendamos que desactive esta opción. Sin embargo, si aún no está familiarizado con las funciones integradas, le recomendamos que active esta opción.
Nivel de ocultación de mensajes	Seleccione hasta qué nivel se ocultarán los mensajes que aparecen. Para que se oculten todos los mensajes, seleccione el nivel más alto posible. Para que todos los mensajes permanezcan visibles de forma permanente, seleccione el nivel más bajo posible.
Tiempo de ocultación de los mensajes	Indique el tiempo de ocultación de los mensajes que desea ocultar en segundos (entre 1 y 60).

4.13.2 Personalizar botones del panel de control

Con estos ajustes, el usuario puede activar y desactivar fácilmente los botones de control que aparecen en el panel de control de la interfaz gráfica de usuario, tanto para la interfaz de escritorio como para la interfaz de panel. El funcionamiento detallado de estos botones se explica en la sección «Panel de control». Es recomendable desactivar un botón si el usuario no desea utilizarlo o no puede hacerlo (por ejemplo, no se puede utilizar el botón de la pinza si no hay ninguna herramienta interactiva conectada), o si el usuario no desea que un operador lo utilice a través de la interfaz de panel.

Mostrar botón *	Actívelo para mostrar este botón y desactívelo para ocultarlo.
Posición de referencia	Especifique los ángulos de la posición de referencia, es decir, la posición vinculada al botón «Homing» accesible en el panel de control. Introduzca un valor para cada ángulo de articulación.

4.13.3 Ajustes de inicio del programa

En esta categoría, todos los ajustes se refieren al proceso de inicio de AX Portal. Al ajustar estos parámetros, se evitan acciones repetitivas durante el arranque. Para algunos ajustes de inicio del programa, seleccione la entrada de la lista «Última selección» (entrada predeterminada) para guardar el ajuste utilizado por última vez y volver a cargarlo.

Proyecto seleccionado	Seleccione el proyecto que se cargará al iniciar el programa. Todos los proyectos definidos aparecen en esta lista.
Aplicación seleccionada	Una vez seleccionado un proyecto, seleccione la aplicación que se cargará al iniciar el programa. Todas las aplicaciones principales definidas para este proyecto aparecen en esta lista.
Evento desencadenante	Seleccione si el robot debe ponerse en marcha automáticamente al iniciar el programa o incluso ejecutar una aplicación. De forma predeterminada, esta opción está desactivada. Para seleccionar un desencadenante, primero deben seleccionarse algunos de los ajustes de inicio mencionados anteriormente, con el fin de evitar que el usuario ejecute aplicaciones no deseadas de forma accidental.

4.13.4 Comportamiento de movimiento

En esta categoría, todos los ajustes se refieren al comportamiento durante los movimientos del robot.

Comportamiento de interrupción	Seleccione el comportamiento de movimiento en caso de interrupción. Encontrará más detalles sobre los distintos comportamientos ante interrupciones y sobre cómo resolverlas en las siguientes subsecciones «Comportamiento ante interrupciones» y «Diálogo de interrupciones».
Configuración de colisión	Establezca el número máximo de intentos para reanudar el movimiento tras producirse una colisión. Si se producen demasiadas colisiones dentro del tiempo de espera definido, el robot interrumpe la reanudación.
Reducción de velocidad en caso de temperatura elevada	Seleccione si la velocidad del robot se reduce automáticamente cuando las temperaturas de los accionamientos son elevadas. Esta configuración está activada de forma predeterminada, lo que hace que los movimientos por encima de los 70 °C se ralenticen automáticamente para evitar el sobrecalentamiento. Si se detectan temperaturas superiores a 90 °C, se activa un error crítico para evitar daños por sobrecalentamiento. En la sección «Límites de rendimiento» del «Manual del robot colaborativo de 6 ejes AX6» se describen más detalles sobre esta función.

4.13.4.1 Comportamiento ante interrupciones

Las interrupciones pueden producirse por diversos motivos, como por ejemplo:

- se ha activado un evento de parada SS1
- se ha activado un evento de parada SS2
- se ha producido una colisión del robot con una persona, un objeto o el entorno
- se ha activado una interrupción del dispositivo de protección

Se detecta una colisión tan pronto como las corrientes de los ejes del robot se desvían del modelo dinámico del robot por encima de un umbral definido. La detección de colisiones es una función de seguridad que ofrece un nivel adicional de protección; no debe utilizarse deliberadamente como parte de una aplicación, ya que las colisiones repetidas someten a una mayor carga a los ejes del robot y aceleran su desgaste.

Tenga en cuenta que, para que la detección de colisiones funcione correctamente, el programador debe ajustar la carga útil del robot al valor correcto tan pronto como se recoge o se deposita un objeto.

Además de las colisiones físicas, se pueden conectar al robot dispositivos de seguridad externos, como barreras fotoeléctricas o dispositivos de protección similares. Estos disparadores actúan de forma preventiva antes de que se produzca un contacto físico, pero se tratan de la misma manera que los eventos de colisión. El comportamiento de interrupción determina cómo el sistema gestiona las interrupciones provocadas tanto por colisiones como por dispositivos de protección.

Actualmente, este es el procedimiento recomendado para el comportamiento de interrupción:

Pausar y Confirmar	Esta configuración hace que la aplicación en ejecución se pause ante una interrupción y espere la confirmación del usuario (ventana de diálogo) para continuar o detener la aplicación.
---------------------------	---

4.13.4.2 Cuadro de diálogo de interrupción

El cuadro de diálogo de interrupción se muestra tan pronto como se produce una condición de interrupción durante el funcionamiento del robot. Los desencadenantes típicos de las interrupciones son colisiones detectadas, la activación de una parada de emergencia, la activación de un dispositivo de protección u otros eventos relevantes para la seguridad. El cuadro de diálogo guía al usuario de forma segura y controlada a través de la resolución de la situación.

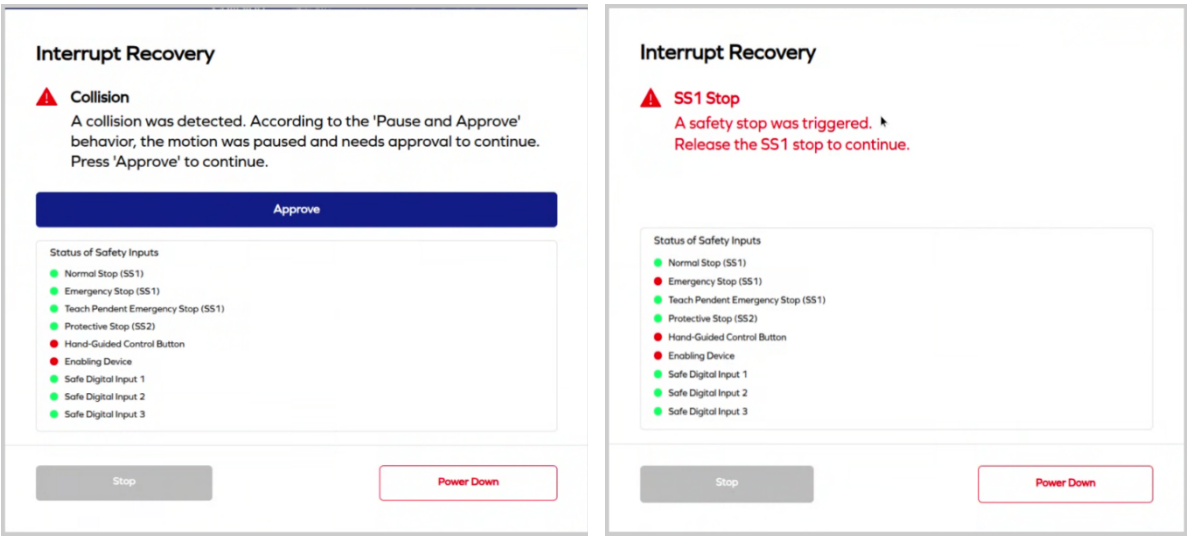


Figura 4.13.4.2: Recuperación tras una interrupción

El cuadro de diálogo se divide en dos áreas principales:

Área superior: información sobre la interrupción y su resolución: esta área describe la condición de interrupción detectada actualmente y proporciona instrucciones claras para su resolución. Algunos ejemplos son:

- Asegurarse de que el robot pueda seguir moviéndose tras una colisión.
- Liberar una parada de emergencia u otros dispositivos de seguridad.
- Restablecer una condición de seguridad, si es necesario.

Dependiendo del tipo de interrupción, se muestra un botón de acción como «Confirmar», «Continuar» o «Restablecer». El usuario debe seguir las instrucciones y pulsar el botón indicado para continuar con el funcionamiento.

Área inferior: estado de las entradas de seguridad: La mitad inferior del cuadro de diálogo muestra el estado en tiempo real de todas las entradas de seguridad relevantes. Esto permite al usuario identificar rápidamente qué dispositivo de seguridad está activo en ese momento, por ejemplo:

- Parada de emergencia (incluida la parada de emergencia en el dispositivo de mando manual)
- Parada de protección
- Dispositivo de autorización
- Pulsador de guiado manual
- Entradas digitales de seguridad

Los indicadores verdes señalan un estado normal, mientras que los indicadores rojos indican un estado activo o de bloqueo (o que no se ha pulsado el botón).

Opción de parada

El botón «Parar» está siempre disponible cuando el usuario debe elegir entre continuar con la aplicación o el movimiento anterior y detenerse. Al pulsar el botón, el robot se detiene y vuelve al estado «Listo», en el que está preparado para nuevas acciones.

Opción de puesta fuera de servicio

El botón «Poner fuera de servicio» está siempre disponible en el cuadro de diálogo de interrupción. Permite apagar el robot en cualquier momento. Tenga en cuenta que el apagado no resuelve automáticamente la condición de interrupción; es posible que sea necesario eliminar la causa subyacente (por ejemplo, una parada de emergencia activada) antes de que se pueda reanudar el funcionamiento normal tras el reinicio.

Flujo de trabajo típico

1. Se produce una interrupción y el cuadro de diálogo se muestra automáticamente.
2. Lea la descripción del estado detectado en la parte superior.
3. Compruebe el estado de las entradas de seguridad en la parte inferior para identificar la causa activa.
4. Solucione la situación manualmente (por ejemplo, desbloquee la parada de emergencia, retire el obstáculo).
5. Pulse el botón de acción que se muestra («Confirmar», «Continuar» o «Restablecer») para continuar.
6. Repita la acción si es necesario hasta que el robot vuelva al funcionamiento normal.

Violación de SLP

Un caso especial de recuperación de interrupción es la violación de SLP (Safe Limited Position), que se produce cuando el robot intenta moverse fuera de los límites de posición establecidos en la configuración de seguridad. En este caso, el cuadro de diálogo de interrupción guía al usuario para

que cambie al modo manual, permitiéndole salir del límite de posición mediante desplazamientos (jogging). A excepción del desplazamiento manual, no se permiten movimientos mientras el robot se encuentre en una infracción de SLP. Si el robot se apaga o se inicia con una infracción de SLP activa, proceda de la siguiente manera:

1. Cambie al modo manual
2. Ponga en marcha el robot
3. Acceda al control de precisión
4. Mueva el robot fuera de sus límites de posición mediante el control de precisión

4.14 Menú de configuración de seguridad

El menú de configuración de seguridad es el punto de acceso al sistema de seguridad del robot de la serie AX. Esta vista está diseñada para **los responsables de seguridad**. Los demás usuarios sólo pueden leer y exportar los parámetros de seguridad, pero no modificarlos.

4.14.1 Interfaz de usuario

Al acceder a los ajustes de seguridad, este menú muestra todos los parámetros de seguridad configurados actualmente en el sistema de seguridad del robot. Cualquier usuario puede ver estos ajustes, pero solo un responsable de seguridad puede modificarlos. Este menú se divide en submenús por los que puede navegar utilizando los números de página o los botones de avance y retroceso situados en la parte inferior de la ventana.

Exportar	Al pulsar el botón «Exportar», la configuración de seguridad se exporta como un archivo *.yaml. Este archivo se puede guardar para realizar una copia de seguridad de la configuración de seguridad y también se puede importar a otros robots para duplicar la configuración de seguridad.
Editar	El responsable de seguridad puede hacer clic en el botón «Editar» (e introducir su contraseña de usuario) para acceder al modo de edición.

En cuanto el menú se encuentra en modo de edición, los ajustes de todos los submenús se pueden modificar. El responsable de seguridad puede ahora ajustar los límites de seguridad de acuerdo con la disposición de montaje y el caso de aplicación del robot. Para facilitar la visualización de los ajustes de seguridad que se han modificado, los campos modificados se marcan en azul.

Importar	Al pulsar el botón «Importar» se puede cargar un archivo previamente exportado con los ajustes de seguridad. De este modo, se sobrescriben los valores modificados manualmente en todos los submenús.
Aplicar	Al pulsar el botón «Aplicar» (e introducir la contraseña de usuario), todos los ajustes de seguridad se envían al sistema de seguridad, que comprueba las opciones y los valores límite. Si la verificación no se realiza correctamente, las entradas no válidas se muestran en rojo en el menú, de modo que se pueden localizar fácilmente. Si la verificación se realiza correctamente, el menú pasa al modo de verificación. Tenga en cuenta que, en este momento, los valores se guardan, pero aún no se verifican. La verificación de la configuración es necesaria para encender el robot.
Cancelar	Al pulsar el botón «Cancelar», no se aplican los ajustes de seguridad y se descartan todos los cambios. A continuación, el menú vuelve al modo de observación (predeterminado).

Tan pronto como el responsable de seguridad haya finalizado la edición de los ajustes de seguridad y se hayan aplicado correctamente todos los valores, este menú se encontrará en modo de verificación. Los ajustes aplicados anteriormente, pero aún no verificados, se marcarán en amarillo.

Verificar	Al pulsar el botón «Verificar página», el responsable de seguridad marca oficialmente como aceptados los ajustes del submenú visible en ese momento y se selecciona el siguiente submenú. Pulsar el botón «Finalizar verificación» del último submenú es el último paso para guardar los ajustes.
------------------	---

Tras comprobar los nuevos valores, es imprescindible validar el comportamiento del sistema de

seguridad. Esto puede realizarse iniciando el sistema en modo de validación de seguridad, lo que permite al software ignorar todos los límites de seguridad durante el proceso de validación para comprobar si el sistema de seguridad se activa según lo previsto.

Safety Settings

Safely Limited Position (SLP)

Defining the areas in the robot's base frame in which the safety settings are applied. ⓘ

Note that a safety margin of 60 mm is subtracted from the specified position limits to determine the effective limits.

	Default		Safe Case 1		Safe Case 2		Safe Case 3		
	min	max	min	max	min	max	min	max	
x	-500	3000	-3000	3000	-3000	3000	-3000	3000	mm
y	-800	800	-3000	3000	-3000	3000	-3000	3000	mm
z	-3000	3000	100	-100	-3000	3000	-3000	3000	mm

Defines whether the limiting area applies to the tool center point (TCP) or all movable parts of the robot.

Limit only the TCP of the robot ☐ Limit all parts of the robot ☒

Tool Safety Zone

Defining a safety zone for the tool and any objects it holds. This zone, represented as a sphere with a specified radius, is offset from the center of the flange frame. This zone cannot enter the workspace boundaries. ⓘ

Tool Safety Offset mm

Tool Safety Radius mm

Cancel

< 1 2 3 4 5 6 >

Import

Apply

Figura 4.14.1: Menú de configuración de seguridad durante la edición de los ajustes

4.14.2 Ajustes de seguridad

La mayoría de los ajustes de seguridad (en todos los submenús) se dividen en 4 columnas. Estas columnas definen los casos de seguridad del robot. Se activa un caso de seguridad cuando se activa la entrada digital de seguridad correspondiente (SDI 1-3). Los parámetros predeterminados están siempre activos, independientemente de las entradas digitales de seguridad.

Las salidas digitales seguras (SDO) no son configurables y sus especificaciones eléctricas se describen en el «Manual del controlador del robot RC-A101».

En algunos ajustes se establece una reserva de seguridad. Este valor indica el rango permitido en el que el software puede activarse antes que el sistema de seguridad para evitar una violación de la seguridad.

4.14.2.1 Posición limitada de seguridad (SLP)

El primer submenú de los ajustes de seguridad contiene la posición limitada de seguridad. Aquí se puede limitar la posición del robot en las direcciones x, y y z, en relación con el sistema de referencia del robot. Si los sistemas de referencia no están claros, al hacer clic en el icono de información se muestra una imagen con las coordenadas. Puede limitar la posición de todas las partes del robot (incluida la base y los articulados) o solo la posición del punto central de la herramienta (TCP) del robot.

Safety Settings

Safely Limited Position (SLP)

Defining the areas in the robot's base frame in which the safety settings are applied. ⓘ

Note that a safety margin of **60 mm** is subtracted from the specified position limits to determine the effective limits.

	Default		Safe Case 1		Safe Case 2		Safe Case 3		
	min	max	min	max	min	max	min	max	
x	-3000	3000	-3000	3000	-3000	3000	-3000	3000	mm
y	-3000	3000	-3000	3000	-3000	3000	-3000	3000	mm
z	-3000	3000	-3000	3000	-3000	3000	-3000	3000	mm

Defines whether the limiting area applies to the tool center point (TCP) or all movable parts of the robot.

Limit only the TCP of the robot ☐ Limit all parts of the robot ☒

Tool Safety Zone

Defining a safety zone for the tool and any objects it holds. This zone, represented as a sphere with a specified radius, is offset from the center of the flange frame. This zone cannot enter the workspace boundaries. ⓘ

Tool Safety Offset mm Tool Safety Radius mm

Export < 1 2 3 4 5 6 > **Edit** **Start Verification**

Figura 4.14.2.1: Ejemplos de ajustes para SLP

4.14.2.2 Ángulos de articulación limitados de forma segura

El segundo submenú de los ajustes de seguridad contiene los ángulos de articulación limitados de forma segura. De forma predeterminada, los ángulos de articulación limitados de forma segura están desactivados. Al activar un caso de seguridad, se puede supervisar la posición angular de las articulaciones.

Sin embargo, dentro de un caso de seguridad, el rango de supervisión de los ángulos de articulación está limitado al intervalo $[-180^{\circ}, 180^{\circ}]$. Esto puede restringir el área de trabajo del robot con respecto a los límites físicos reales del mismo. Para evitar esta restricción, la supervisión de los ejes 1, 4 y 6 se puede desactivar de forma individual.

Los ángulos de articulación con límite de seguridad solo están activos si se definen tanto el límite inferior como el superior. Si se dejan vacíos ambos límites, se desactiva la supervisión para la articulación correspondiente. Tenga en cuenta que los límites inferior y superior deben dejarse vacíos o definirse ambos.

Safety Settings

Safely Limited Joint Angles

Defining the position limits of each joint of the robot. ⓘ

The default position limits are determined by the physical constraints of the joints and cannot be modified.

Note that some joints can move beyond **180 deg**, but safe cases do not support these ranges. Disabling the limits allows the full joint range to be used.

Note that a safety margin of **5 deg** is subtracted from the specified position limits to determine the effective limits.

	Default		Safe Case 1		Safe Case 2		Safe Case 3		
	min	max	min	max	min	max	min	max	
Joint 6	Disabled		Dis.	Dis.	Dis.	Dis.	Dis.	Dis.	deg
Joint 5	Disabled		-160	160	-160	160	-160	160	deg
Joint 4	Disabled		Dis.	Dis.	-100	100	Dis.	Dis.	deg
Joint 3	Disabled		-165	165	-165	165	-165	165	deg
Joint 2	Disabled		-160	160	-160	160	-160	160	deg
Joint 1	Disabled		Dis.	Dis.	Dis.	Dis.	-179.9	179.9	deg

Export

< 1 2 3 4 5 6 >

Edit

Start Verification

Figura 4.14.2.2: Ejemplos de ajustes para ángulos de articulación limitados de forma segura

4.14.2.3 Velocidad limitada de forma segura (SLS)

El tercer submenú de los ajustes de seguridad contiene la velocidad limitada de forma segura. Aquí se puede limitar individualmente la velocidad de cada eje del robot, así como la velocidad de movimiento de todas las partes del robot, incluidas la herramienta, la base y los articulados.

Safety Settings

Safely Limited Speed (SLS)

Defining the speed limits of each joint and all movable parts of the robot.

Note that a safety margin of **7 deg/s** or **100 mm/s** is subtracted from the specified speed limits to determine the effective limits.

	Default	Safe Case 1	Safe Case 2	Safe Case 3	
Joint Velocity 6	247	247	247	247	deg/s
Joint Velocity 5	247	247	247	247	deg/s
Joint Velocity 4	247	247	247	247	deg/s
Joint Velocity 3	187	187	187	187	deg/s
Joint Velocity 2	187	187	187	187	deg/s
Joint Velocity 1	187	187	187	187	deg/s
Speed	2000	2000	2000	2000	mm/s

Export

<

1

2

3

4

5

6

>

Edit

Start Verification

Figura 4.14.2.3: Ejemplos de ajustes para SLS

4.14.2.4 Ajustes de seguridad para potencia y fuerza

El cuarto submenú de los ajustes de seguridad contiene ajustes relativos a la potencia y la fuerza:

- Para cada caso de seguridad, el usuario puede establecer un valor límite de fuerza diferente, que define la sensibilidad con la que el robot reacciona ante fuerzas externas, como colisiones con personas u objetos. El valor límite de fuerza se aplica en el punto central de la herramienta (TCP).
- Es importante establecer la carga útil media (y los valores correctos de carga útil durante el funcionamiento), definida como el peso de la herramienta más los objetos transportados, para ajustar los límites de fuerza de forma más precisa a su aplicación específica.
- El vector de gravedad del robot debe ajustarse a la forma de montaje del robot. Si el robot está montado en el suelo, el vector de gravedad es $[0, 0, -9,81]$. Sin embargo, si el robot está montado en una pared, en una superficie inclinada o en el techo, primero debe calcularse el vector de gravedad e introducirse aquí correctamente.

Safety Settings

Safe Power and Force Limit (PFL)

Defining the maximum allowable force at the tool center point (TCP). If this limit is exceeded, a collision interrupt is triggered. Raising the limit reduces collision detection sensitivity. When payload variations are large, the force limit may need to be increased to account for calculation differences.

	Default	Safe Case 1	Safe Case 2	Safe Case 3	
Force Limit	250	250	250	250	N

Average Payload

Defining the robot's average payload — the combined weight of its tool and any picked items. During operation, the payload changes as the robot picks up and places items. The safety system uses the average payload to assess whether the applied forces remain within acceptable limits.

Average Payload: 0 kg

Gravity Vector

Defining the gravity vector in the robot's base frame based on its mounting orientation.

	x	y	z	
Gravity Vector	0	0	-9.81	m/s ²

Export < 1 2 3 4 5 6 > Edit Start Verification

Figura 4.14.2.4: Ajustes seguros para los límites de potencia y fuerza

4.14.2.5 E/S de seguridad y restricciones de usuario

El quinto submenú del menú de ajustes de seguridad permite configurar el comportamiento de las E/S relevantes para la seguridad y los permisos de los operadores. Los usuarios pueden definir cómo se gestionan las paradas de seguridad, si se supervisan las entradas en modo manual, las condiciones de reinicio, los permisos para el cambio de modo de funcionamiento, así como los requisitos de contraseña para los cambios de funcionamiento.

The screenshot shows a window titled "Safety Settings" with a blue header bar. The window is divided into two main sections: "Safe I/O Configuration" and "Operator Limitations".

Safe I/O Configuration

Defining whether a protective stop input results in a SS1 or SS2 stop.

Results in a SS2 Stop ☐ Results in a SS1 Stop ☒

Defining whether a protective stop input is always being normally observed, or if it is to be ignored during manual mode.

Ignore in manual mode ☐ Observe in manual mode ☒

Defining whether resetting interrupts can only be triggered via digital inputs.

Trigger always ☐ Digital inputs only ☒

Operator Limitations

Defining whether operators are able to switch the operation mode.

Allow switching mode ☐ Don't allow switching mode ☒

Defining whether operators need to enter their password to switch the operation mode.

No password required ☐ Password required ☒

At the bottom of the window, there is a navigation bar with an "Export" button, a set of numbered tabs (1-6) with tab 5 selected, an "Edit" button, and a "Start Verification" button.

Figura 4.14.2.5: Ajustes de configuración para E/S de seguridad

4.14.2.6 Datos de verificación

El último submenú de los ajustes de seguridad contiene información sobre la configuración de seguridad:

- Las sumas de comprobación leídas de la configuración de seguridad (para la resolución de errores)
- El contador de transferencias (indica cuántas veces se han escrito los parámetros de seguridad)
- Cuándo se actualizaron por última vez los parámetros de seguridad
- Si se han verificado los parámetros de seguridad
- El número de serie y la versión del firmware del dispositivo de seguridad

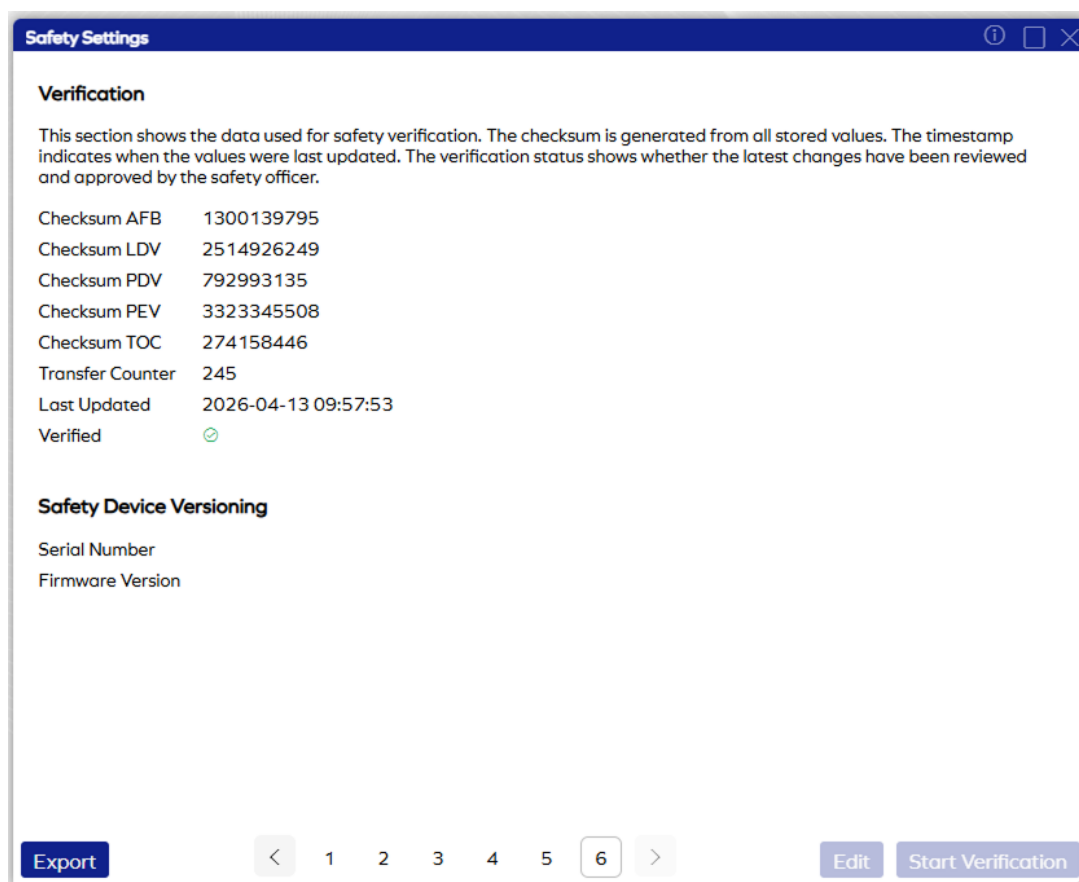


Figura 4.14.2.6: Ajustes de verificación

4.14.2.7 Funciones de salida de seguridad

Los estados lógicos de determinadas funciones de seguridad están conectados a las salidas de seguridad del controlador del robot. Encontrará información detallada en el «Controlador del robot RC-A101: Manual».

4.14.2.8 Conexión de dispositivos opcionales

Para la conexión de dispositivos opcionales, como el soporte para tableta o el interruptor de autorización, consulte el «Manual del controlador del robot RC-A101».

5. Interfaz del panel (GUI)

5.1 Introducción

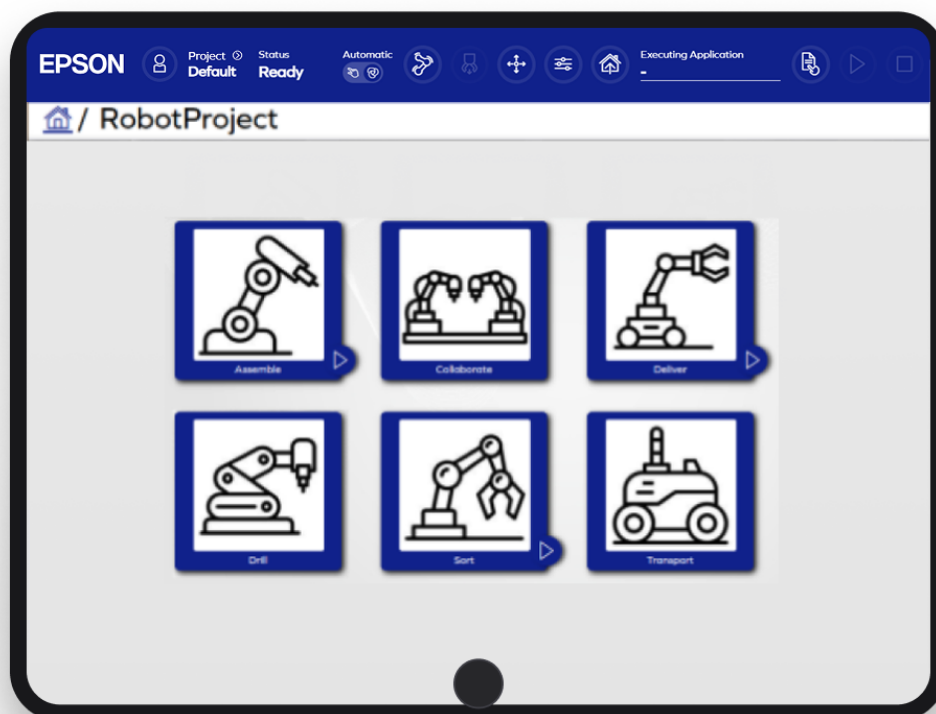


Figura 5.1.a: Captura de pantalla de ejemplo de la interfaz del panel

La interfaz del panel es una alternativa a la interfaz de escritorio que permite acceder rápidamente a sus aplicaciones principales. Esta interfaz se ha diseñado para que sea totalmente personalizable según los requisitos del cliente y, en particular, para que muestre exclusivamente las aplicaciones (y la información sobre su estado). La interfaz del panel es compatible con pantallas táctiles y su diseño se asemeja a la conocida estructura tipo aplicación de los dispositivos móviles modernos, lo que la hace intuitiva incluso para usuarios sin conocimientos técnicos. Además, con la interfaz del panel, el usuario no tiene que preocuparse por la programación ni por el código que hay detrás de la aplicación. Basta con hacer clic en la aplicación correspondiente para que esta se inicie. Se puede acceder a esta interfaz a través de la interfaz de administración iniciando sesión y utilizando el botón «Go to Panel Interface».

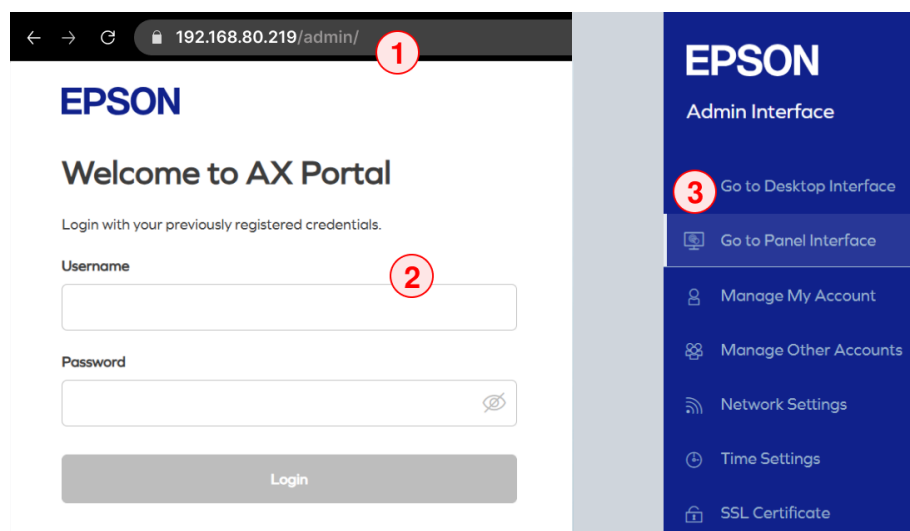


Figura 5.1.b: Acceso a la interfaz del panel

En algunos navegadores (especialmente en Windows), el escalado o la ampliación de la pantalla están configurados de forma predeterminada en un valor alto (por ejemplo, 150 %), lo que puede provocar que algunas partes de la interfaz (como el Cockpit o el logotipo) aparezcan recortadas. Para obtener la mejor visualización, recomendamos ajustar la escala de visualización al 100 % (o incluso menos).

5.2 Configuración de aplicaciones para la interfaz del panel

De forma predeterminada, la interfaz del panel no muestra ninguna aplicación. El usuario debe especificar primero qué aplicación debe mostrarse allí para que el operador pueda ejecutarla rápidamente. Para configurar una aplicación del panel, proceda de la siguiente manera:

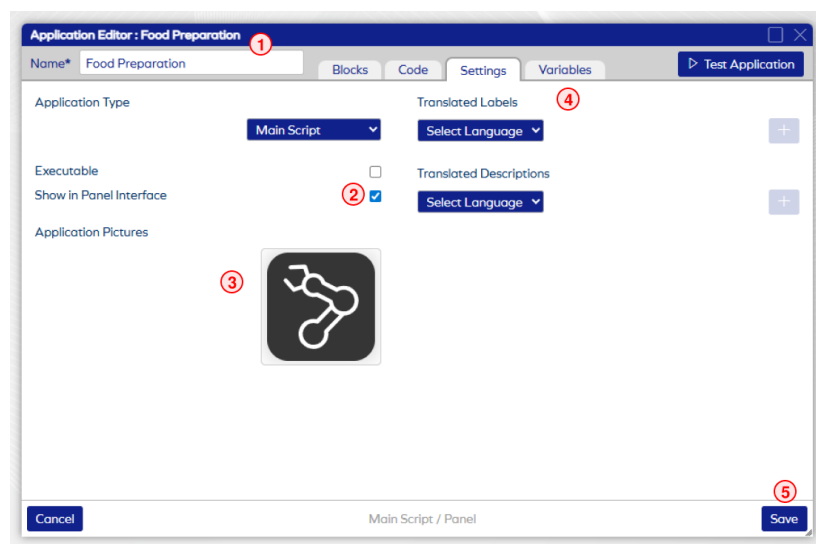


Figura 5.2.a: Ajustes relevantes para la interfaz del panel en el editor de aplicaciones (interfaz de escritorio)

Ajustes relevantes para la interfaz del panel: 2., 3., 4.

1. Vaya a la interfaz de escritorio, edite la aplicación y abra los ajustes de la aplicación;

2. active la opción «Mostrar en la interfaz del panel»,
3. cargue opcionalmente una imagen de aplicación adecuada (la relación de aspecto óptima es 5:4), que se mostrará más tarde junto con el nombre de la aplicación en el menú principal de la interfaz del panel,
4. opcionalmente, vaya a la pestaña «Variables de aplicación» para añadir variables de estado con las que podrá supervisar y/o modificar variables a través de la interfaz del panel,
5. y, por último, guarde la aplicación.

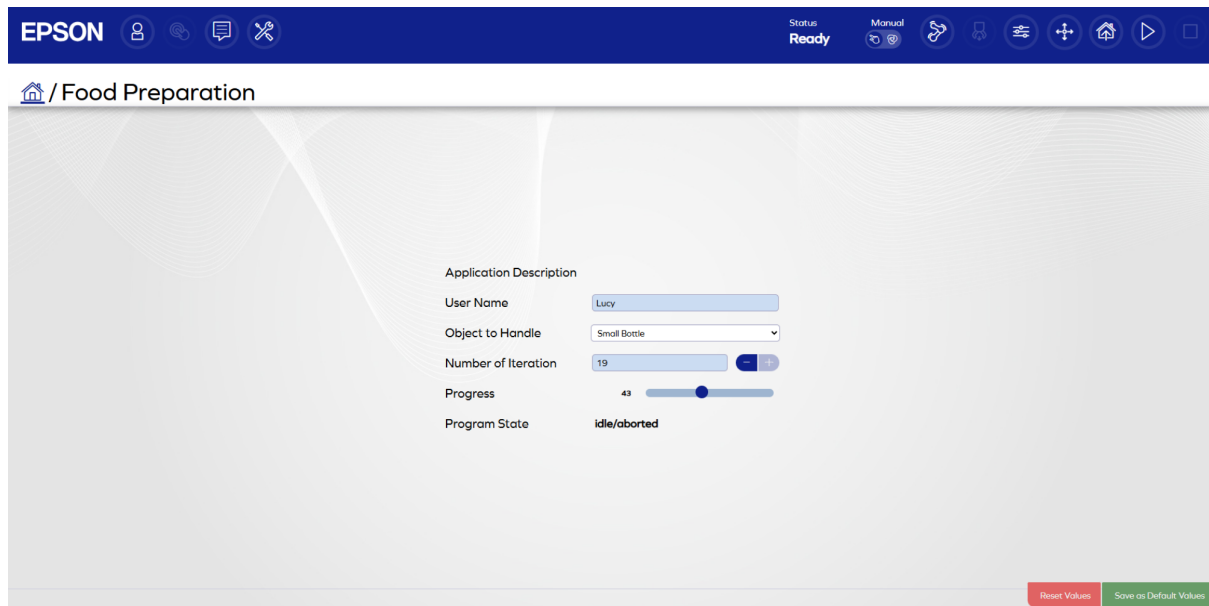


Figura 5.2.b: Aplicación de panel con variables de estado

5.3 Panel de control y ajustes

El panel de control de la interfaz del panel muestra, en gran medida, los mismos botones que el panel de control de la interfaz de escritorio. Encontrará una explicación detallada de todos los elementos en la sección Panel de control. Además de estos elementos, verá en el lado izquierdo algunos botones diferentes:

	El botón «Panel» abre la página principal del panel de control, en la que se muestran las aplicaciones que se pueden seleccionar y ejecutar.
	El botón «Mensajes» abre el historial de mensajes, en el que se muestran todos los mensajes anteriores, incluidas las advertencias y los errores, en orden cronológico.
	El botón «Configuración» le lleva a la ventana de configuración, que contiene únicamente los ajustes más importantes para un acceso rápido, a saber • selección de la herramienta conectada, • puesta en marcha y parada del robot, • descarga de los archivos de registro.

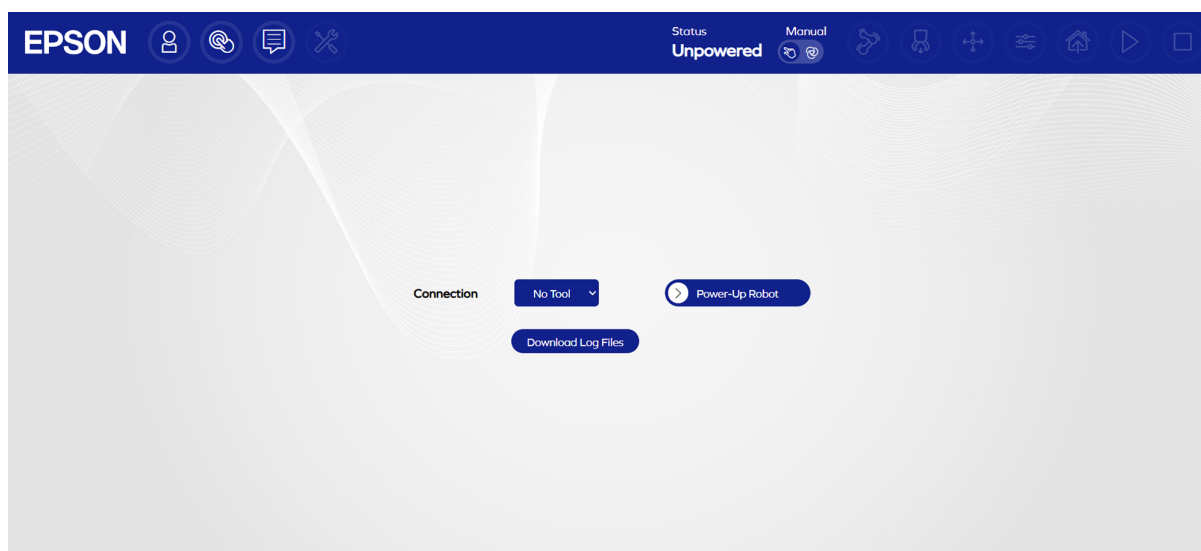


Figura 5.3: Ajustes de la interfaz del panel

5.4 Control de aplicaciones

En la ventana principal de la interfaz del panel se enumeran todas las aplicaciones en una estructura similar a la de una aplicación, mostrando su nombre y su imagen. El panel distingue entre dos tipos de aplicaciones:

- Las aplicaciones sin variables de estado muestran un botón de reproducción. Pulse sobre la aplicación para ejecutarla directamente.
- Las aplicaciones con variables de estado se muestran sin botón de reproducción. Pulse sobre las aplicaciones para abrir la vista detallada con todas las variables de estado. En esta ventana puede modificar las variables de estado editables y pulsar el botón de ejecución para ejecutar estas aplicaciones. Pulse el botón Atrás para volver a la ventana principal del panel.

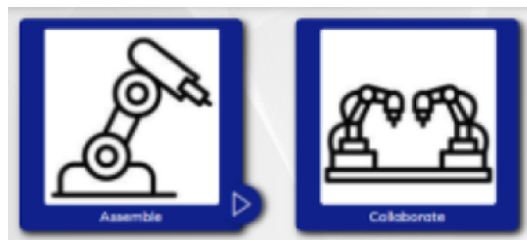


Figura 5.4: Botones para la selección de aplicaciones

6. Definiciones cinemáticas

La cinemática del robot describe cómo se mueve el robot y cómo se pueden representar sus movimientos. Para trabajar con la cinemática, el usuario debe comprender las diferentes definiciones de los parámetros cinemáticos y los objetos disponibles:

6.1 Coordenadas

Las coordenadas definen una posición y orientación absolutas del robot. La posición es un vector con tres valores $[x, y, z]$ en mm, y la orientación es también un vector con tres valores $[\text{roll}, \text{pitch}, \text{yaw}]$ (ángulos náuticos) en grados.

La inicialización puede realizarse de la siguiente manera:

```
coord = Coordinates([600, 0, 300, 180, -90, 0])
# ... this is similar to ...
coord = Coordinates({"x": 600, "y": 0, "z": 300, "roll": 180, "pitch": -90, "yaw": 0})
```

El acceso a los elementos puede realizarse de la siguiente manera:

```
coord = Coordinates([600, 0, 300, 180, -90, 0])

# accessing individual elements (x, y, z, roll, pitch, yaw)
x = coord.x
x = coord["x"]
x = coord[0]

# accessing position or orientation
position = coord.position
orientation = coord.orientation
```

6.1.1 Ángulos náuticos

Los ángulos náuticos son una de las muchas formas de definir la parte de orientación (balanceo, cabeceo, guiñada) de las coordenadas. Los ángulos náuticos se utilizan habitualmente para la orientación de aeronaves. En los ángulos náuticos, una rotación tridimensional se define como

1. una rotación alrededor del eje z negativo en la dirección de guiñada (yaw, ψ),
2. una rotación alrededor del eje y negativo en la dirección de cabeceo (pitch, θ) y
3. una rotación alrededor del eje x positivo en la dirección de balanceo (balanceo, ϕ).

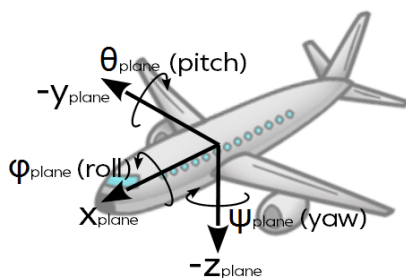


Figura 6.1.1.a: Sistema de referencia náutico

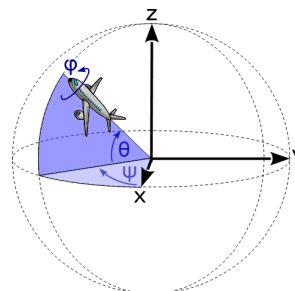


Figura 6.1.1.b: Orden de las rotaciones

Nota: Tan pronto como una función requiera que trabaje con orientaciones tridimensionales del efector final, resulta útil imaginar un plano fijado al efector final, de modo que los sistemas de referencia del efector final y de la aeronave coincidan. De este modo, los ángulos (balanceo, cabeceo y guiñada) que conducen a esa orientación concreta se pueden deducir con mucha mayor facilidad.

6.1.2 Sistemas de referencia

Para la manipulación de los robots de la serie AX, son importantes los siguientes sistemas de referencia:

- El **sistema base** de un robot se define de tal manera que su eje z apunta alejándose de la placa base y el eje x discurre paralelo a la placa base y apunta alejándose de las conexiones de la base. La conocida regla de la mano derecha define entonces la dirección del eje y.
- El **sistema de trabajo** de un robot es un sistema de referencia definido por el usuario en relación con el sistema de referencia base y se corresponde con este, salvo que se indique lo contrario. Las coordenadas se especifican en este sistema de referencia. El usuario puede utilizar este sistema de referencia para definir su propio espacio de trabajo para el robot.
- El **sistema de brida** de un robot varía en función del tipo de herramienta:
 - o Si no hay ninguna herramienta montada, el origen de este sistema de referencia se encuentra en el centro de la brida (placa de conexión) en la parte superior del robot. La orientación coincide con la del sistema base cuando todos los ángulos de articulación son cero.
 - o Si se ha montado una herramienta definida por el usuario (y se ha definido mediante el editor de herramientas), el sistema de referencia se desplaza a la posición y orientación especificadas en dicha herramienta.
- El **sistema de herramientas** de un robot es un sistema de referencia definido por el usuario en relación con el sistema de referencia de la brida y coincide con este, salvo que se indique lo contrario. Las coordenadas se obtienen en el origen de este sistema de referencia. El usuario puede utilizar este sistema de referencia para definir el punto central de la herramienta.

En la siguiente ilustración se muestra un ejemplo de estos sistemas de referencia:

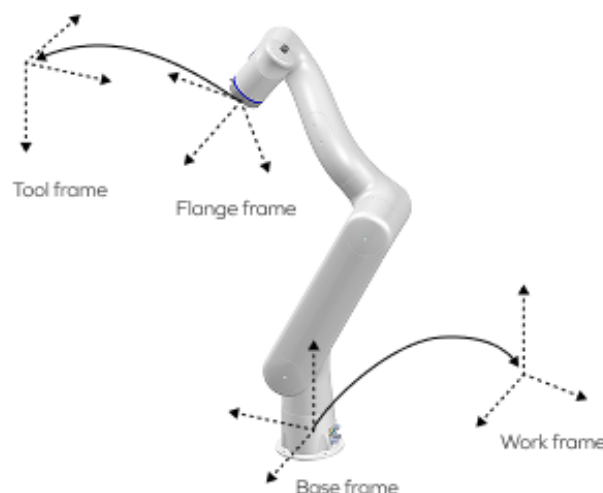


Figura 6.1.2: Sistemas de referencia del robot

6.2 Configuración

Para una posición y orientación dadas del robot, la cinemática inversa puede proporcionar varias configuraciones de articulaciones diferentes. Estas configuraciones se diferencian en la forma en que se angulan o giran las articulaciones. Con el ajuste de configuración se selecciona una configuración concreta de entre estas opciones posibles. En un robot típico de seis ejes, las diferentes combinaciones de las direcciones de los hombros, los codos y las muñecas pueden dar lugar a hasta ocho configuraciones diferentes, incluso si la posición y la orientación del efector final son idénticas. Para distinguirlas, estas configuraciones se denominan «c1» a «c8».

Tenga en cuenta que las configuraciones no son adecuadas para comparar dos poses o para comprobar si una trayectoria es continua. Un cambio en la configuración no hace que una trayectoria sea necesariamente discontinua. Del mismo modo, una trayectoria discontinua no tiene por qué contener necesariamente cambios de configuración.

6.3 Postura

Una pose define las coordenadas y la configuración del robot. Se puede establecer mediante el ID o el nombre de una pose predefinida en la base de datos. Alternativamente, se puede crear especificando las coordenadas y la configuración mediante la función «Crear nueva pose» o indicando los ángulos de las articulaciones mediante la función de cinemática directa.

La inicialización puede realizarse de la siguiente manera:

```
# read predefined pose from database by index
pose = Pose(1)
# read predefined pose from database by name
pose = Pose("fancy_pose")
```

6.4 Punto

Un punto es un elemento discreto de una trayectoria. Además de la pose del robot, indica la velocidad del robot y el intervalo de tiempo transcurrido desde el punto anterior. Esto se denomina PVT (posición, velocidad, tiempo) y suele almacenarse como un vector de ángulos de articulación, un vector de velocidades de articulación y un intervalo de tiempo (flotante). Tenga en cuenta que una versión ampliada de un punto puede incluir además la aceleración del robot.

6.5 Trayectoria

Una trayectoria define cómo se desplaza el robot entre dos o más posiciones. Una trayectoria continua se define exclusivamente en el espacio de herramientas mediante varios segmentos, que pueden ser lineales o circulares. A continuación, la trayectoria se discretiza y se traslada al espacio de articulaciones. Una trayectoria se denomina «sin discontinuidades» cuando las soluciones discretas de la trayectoria no requieren un reposicionamiento del robot (cambio de configuración). Una trayectoria se denomina continua cuando la aceleración a lo largo de la misma es constante.

Una trayectoria comienza en una pose especificada (a diferencia de las coordenadas). Los segmentos de la trayectoria suelen definir coordenadas, ya que la configuración de la trayectoria viene determinada por la pose inicial. Al utilizar la solución más cercana de la cinemática inversa al punto

discreto anterior, la trayectoria se vuelve automática y sin discontinuidades siempre que sea posible. Tenga en cuenta que utilizamos los términos «trayectoria» y «trayecto» como sinónimos, aunque, desde un punto de vista técnico, una trayectoria no contiene información sobre la velocidad.

La inicialización puede realizarse de la siguiente manera:

```
# read predefined path from database by name
path = Path("fancy_path")
```

6.6 Segmento

Un segmento representa una parte individual de una ruta que define cómo se desplaza el TCP hacia el destino.

Los tipos de segmento disponibles son:

- **Lineal:** el TCP se desplaza en línea recta hacia las coordenadas de destino definidas por el segmento.
- **Circular:** el TCP sigue una trayectoria circular desde su posición actual, pasando por una posición intermedia definida, hasta las coordenadas de destino del segmento. Tenga en cuenta que la orientación del TCP durante este movimiento la determina automáticamente el algoritmo.
- **Coordenadas:** el TCP se desplaza por la ruta más eficiente hacia las coordenadas de destino. Esto corresponde a un movimiento lineal en el espacio articular.
- **Postura:** El TCP se desplaza por la trayectoria más eficiente hacia la postura de destino. Esto equivale a un movimiento lineal en el espacio articular.
- **Orientación:** El TCP realiza un movimiento de rotación puro para alcanzar la orientación de destino especificada, mientras mantiene su posición actual.

Para cada segmento se pueden definir parámetros de velocidad y aceleración:

- Para los segmentos de orientación, estos corresponden a la velocidad angular en [deg/s] y a la aceleración angular en [deg/s²]
- Para todos los demás tipos de segmentos, corresponden a la velocidad lineal en [mm/s] y a la aceleración lineal en [mm/s²]

Además, se puede establecer un radio de transición (Blend Radius) que define la desviación máxima permitida en mm respecto a la pose final del segmento anterior, que corresponde a la pose inicial del segmento actual, durante la transición. Un radio de transición de 0 mm obliga al TCP a recorrer exactamente la pose definida, mientras que un valor mayor permite transiciones más suaves y naturales, al permitir que la trayectoria entre los segmentos sea curva.

6.7 Rejillas

Las rejillas son matrices de puntos personalizables, de hasta tres dimensiones, que, por ejemplo, definen las posiciones de bandejas predefinidas y pueden utilizarse para agrupar y almacenar conjuntos de valores de coordenadas. Por lo general, una rejilla se interpreta como una matriz bidimensional; sin embargo, también puede ser una simple lista de puntos sin una segunda dimensión

(por ejemplo, para recoger objetos de una cinta transportadora) o incluso una estructura tridimensional (por ejemplo, para apilar objetos en cajas).

7. Módulos de conexión

Además de las interfaces de usuario intuitivas y fáciles de usar para el control y la supervisión del robot, AX Portal también incluye una serie de módulos de conexión que permiten controlar y supervisar el robot a través de diversos métodos de comunicación, como TCP, REST, ROS y Modbus. Cada uno de estos métodos de comunicación es adecuado para diferentes casos de uso:

- **TCP** es ideal para la comunicación en tiempo real, en la que se requiere un intercambio de datos rápido y continuo. Se utiliza con frecuencia para aplicaciones de control directo, implementaciones de clientes personalizadas o para la integración en sistemas que requieren conexiones de socket persistentes.
- **REST** es el más adecuado para la integración en servicios o aplicaciones basados en la web. Se suele emplear cuando la facilidad de uso, la escalabilidad y la compatibilidad con sistemas estandarizados basados en HTTP son más importantes que el rendimiento en tiempo real.
- **ROS** (Robot Operating System) está diseñado para la investigación y el desarrollo en el ámbito de la robótica. Resulta especialmente útil para la integración del robot en sistemas robóticos complejos y permite funciones como la fusión de sensores, el comportamiento autónomo y la interacción con otros dispositivos compatibles con ROS.
- **Modbus** se utiliza con frecuencia en entornos de automatización industrial. Es muy adecuado para la comunicación con PLC (controladores lógicos programables) y otros dispositivos industriales en los que la fiabilidad y la estandarización son fundamentales.

En este capítulo se explica el uso de estos módulos.

7.1 Conexiones, autenticación y control activo

Antes de utilizar un módulo de conexión, es importante que comprenda cómo se conectan estos módulos con los clientes, cómo funciona la autenticación y cómo funciona el principio del control activo en AX Portal.

7.1.1 Canales de conexión

La conexión entre un dispositivo y el robot se realiza siempre a través de un canal de conexión específico. Estos canales de conexión se implementan de forma ligeramente diferente para cada uno de los distintos módulos de conexión.

El **módulo TCP** admite **varios canales**. Esto significa que cada dispositivo que desee conectarse al robot crea su propio canal de conexión y se registra por separado. El cierre de uno de estos canales interrumpe la conexión y, además, libera el control activo (véase más abajo).

 ADVERTENCIA	Los módulos REST, ROS y Modbus disponen cada uno de un único canal. Esto significa que, tan pronto como se establece y autentifica la conexión con uno de estos módulos, cualquier dispositivo puede enviar comandos al robot. Estos canales se crean al iniciar AX Portal y no pueden cerrarse al conectar dispositivos.
---	--

7.1.2 Autenticación

AX Portal utiliza una autenticación segura de dos pasos. En primer lugar, siga el procedimiento de inicio de sesión remoto descrito en la sección Autenticación segura para obtener un token de acceso

válido. Este paso debe realizarse siempre a través de una **conexión TCP independiente**. Una vez que haya iniciado sesión correctamente y haya generado un token de acceso, puede registrar dicho token a través del canal de conexión seleccionado con el comando **token_login** para autenticar a su usuario.

 PRECAUCIÓN	El paso de autenticación se puede omitir si desactiva la autenticación a través de TCP en el menú de configuración. Tenga en cuenta que el módulo Modbus no ofrece la posibilidad de enviar un comando token_login; por lo tanto, la autenticación de este módulo debe desactivarse en el menú de configuración para poder utilizarlo.
--	--

7.1.3 Control activo

AX Portal cuenta con una protección de la estación de control activa, de modo que otros usuarios o conexiones no puedan controlar el robot al mismo tiempo. Sin esta protección, podrían surgir situaciones potencialmente peligrosas.

El comando **request_active_control** es utilizado por un canal de conexión para convertirse en la estación de control activa. Una vez que este comando se ha ejecutado con éxito, dicha estación de control mantiene el control de forma permanente hasta que se cierre el canal o el robot quede sin tensión. Si otra estación de control tiene el control, este comando queda bloqueado hasta que dicha estación de control acepte la solicitud de control.

El comando **release_active_control** es utilizado por la estación de control activa para ceder el control activo. Se recomienda liberar siempre el control tan pronto como el canal deje de utilizarse activamente. De lo contrario, este canal bloqueará el control activo de forma permanente.

7.2 Módulo de conexión TCP

7.2.1 Establecer conexión

El lado del servidor del módulo de conexión TCP está basado en [Python](#) y se ha implementado utilizando la [biblioteca de sockets](#) integrada. De forma predeterminada, se puede acceder al servidor a través del **puerto 38152**. El mensaje está codificado con el estándar **UTF-8**. Los mensajes se separan mediante el carácter de fin de texto 0x03 (en Python «\x03»). Utilice el siguiente script de ejemplo para conectarse a su robot y mostrar todas las respuestas entrantes. El significado de los mensajes entrantes se explica más adelante en la sección «Recepción de comandos».

```
import socket
import time

# define the robot ip address
ip_address = "10.0.0.203"

# create the tcp socket
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as client_socket:
    # connect to the server
    client_socket.connect((ip_address, 38152))
    # prepare message buffer
    message_buffer = ""
```

```
while True:
    # read data and add it to message buffer
    try:
        message_buffer += client_socket.recv(1024).decode("UTF-8")
    except socket.timeout:
        time.sleep(0.1)
        continue
    # check for complete messages in buffer
    if "\x03" in message_buffer:
        # extract complete messages
        message_split = message_buffer.split("\x03")
        for message in message_split[:-1]:
            print(message)
        # incomplete message (if any) remains in buffer
        message_buffer = message_split[-1]
```

7.2.2 Envío de comandos

Antes de enviar comandos al robot, asegúrese de haber comprendido los principios explicados en la sección Conexiones, autenticación y control activo. El siguiente ejemplo envía los comandos iniciales **token_login** y **request_active_control**.

```
import json
import socket
import time

# define the robot ip address, access token, and session identifier
ip_address = "10.0.0.203"
token = "MY_ACCESS_TOKEN"
session_id = "MY_SESSION_ID"

# create the tcp socket
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as client_socket:
    # connect to the server
    client_socket.connect((ip_address, 38152))

    # use the authentication token to log in (needed if authentication is not disabled)
    request = {"action": "token_login", "bridge": "core",
               "arguments": {"token": token, "session_id": session_id}}
    message = json.dumps(request) + "\x03"
    client_socket.sendall(message.encode("UTF-8"))

    time.sleep(0.5)

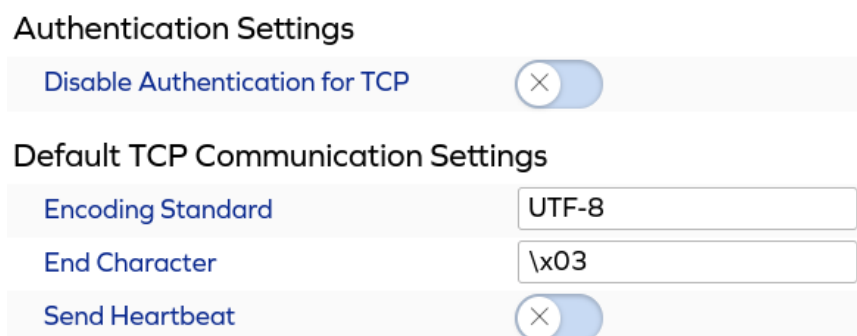
    # make sure to acquire active control, which makes the TCP connection the active
    # control station of the robot (only one control station can be active at any time)
    request = {"action": "request_active_control", "bridge": "core",
               "arguments": {}}
    message = json.dumps(request) + "\x03"
    client_socket.sendall(message.encode("UTF-8"))

    time.sleep(0.5)

# add more robot control commands here
```

7.2.3 Configuración

Los ejemplos anteriores utilizan la siguiente configuración TCP predeterminada, que se puede modificar en el menú de configuración de la interfaz de escritorio. Aquí puede desactivar de forma permanente la autenticación para TCP, así como cambiar la codificación y el carácter de fin de los mensajes TCP. Además, puede activar un «heartbeat» para comprobar continuamente si la conexión sigue activa.



The image shows a configuration window with two sections. The first section, 'Authentication Settings', contains a toggle switch for 'Disable Authentication for TCP' which is currently turned on. The second section, 'Default TCP Communication Settings', contains three fields: 'Encoding Standard' set to 'UTF-8', 'End Character' set to '\x03', and a toggle switch for 'Send Heartbeat' which is currently turned off.

Authentication Settings	
Disable Authentication for TCP	☒

Default TCP Communication Settings	
Encoding Standard	UTF-8
End Character	\x03
Send Heartbeat	☐

Figura 7.2.3: Configuración del módulo TCP

7.2.3.1 Estándar de codificación

Aquí puede definir la codificación de los mensajes enviados a través de la conexión TCP. Se aceptan los [estándares de codificación](#) habituales Python . Los estándares más utilizados son UTF-8 (valor predeterminado) o ASCII.

7.2.3.2 Carácter de fin

Aquí puede cambiar el carácter de fin de los mensajes enviados a través de la conexión TCP. El carácter de fin indica el final del mensaje. Al recibir un mensaje, el carácter de fin se elimina automáticamente del mensaje.

7.2.3.3 Enviar latido

Aquí puede especificar si el servidor TCP envía automáticamente un latido. Se envía un latido cada 50 ms y consiste en un mensaje vacío (se envía el carácter de fin).

7.2.4 Protocolo de mensajes: recepción de comandos

Todos los comandos enviados y recibidos son cadenas [JSON](#) . Como se ha indicado anteriormente, los mensajes se codifican con el estándar de codificación UTF-8 y se separan mediante el carácter de fin de texto 0x03 (en Python «\x03»). Un comando recibido es un diccionario codificado en JSON que contiene una clave (key) y un valor. La clave define el tipo de mensaje. El valor contiene la información del mensaje. En la sección «Establecer conexión» encontrará un ejemplo de código que recibe estos mensajes del robot.

Tenga en cuenta que esta lista no es exhaustiva, pero contiene los comandos más importantes.

7.2.4.1 Información de estado

El estado del robot se envía tan pronto como cambia.

Clave	Valor	Tipo
status_info	El estado actual del robot según la siguiente lista: ● 0 → Fuera de servicio ● 1 → Conectando ● 2 → Cargando ● 3 → Listo ● 4 → En proceso ● 5 → Calibración ● 6 → En funcionamiento ● 7 → Modo autónomo ● 8 → En pausa ● 9 → Se reanudará ● 10 → Parada de seguridad ● 11 → Parada de seguridad ● 12 → Límite de posición ● 13 → Control manual ● 14 → Colisión ● 15 → Parada de emergencia ● 16 → Parada normal ● 17 → Fallo ● 18 → Desconexión ● 19 → Detenido	Entero

```
{"status_info": 0}
```

7.2.4.2 Valor de retorno de la acción

Tras la ejecución de una acción, el valor de retorno se envía junto con el comando de la acción. Si la acción no tiene valor de retorno, se envía como nulo. La acción solo se considera exitosa si no se ha producido ningún error durante la ejecución de la función. Por último, el número de secuencia es un parámetro opcional que solo se añade si el usuario ha especificado una secuencia al enviar el comando.

Key	Valor	Tipo
action_return_value	El nombre de la acción de retorno, el valor de retorno y si la acción se ha completado con éxito. En caso de error, la acción no se ha completado con éxito.	Diccionario

```
{"action_return_value": {"action": "release", "value": null,
                        "sequence": 100, "success": true}}
```

7.2.4.3 Errores

Si se produce un error, se envía un mensaje de error con información más detallada.

Clave	Valor	Tipo
error	La información del error.	Diccionario

```
{"error": {"text": "The argument 'variable_value' is missing for the
              'set_state_variable' action.",
          "title": "Minor Error",
          "color": 3,
          "code": "e0010035",
          "time": 1554907874.0436678}}
```

7.2.4.4 Historial de mensajes

Lista de mensajes anteriores enviados a la interfaz de usuario.

Clave	Valor	Tipo
message_history	Los mensajes contienen texto, título, código y marca de tiempo.	Lista

```
{"message_history": [{"text": "The specified bridge does not exist.",
                        "title": "Minor Error", "color": 3,
                        "code": "e0010019",
                        "time": 1554897027.5446029}]}
```

7.2.4.5 Definición de un grupo de estados

Define los datos de posición del robot. Un grupo de estados contiene un ID que indica a qué se refieren los datos incluidos. Los datos pueden identificarse a partir de las definiciones.

Clave	Valor	Tipo
state_group_define	Definición del grupo de estados.	Diccionario

```
{"state_group_define": {"id": "pose",
  "label": "tcp_pose",
  "members": [{"id": "x", "label": "x"},
    {"id": "y", "label": "y"},
    {"id": "z", "label": "z"},
    {"id": "roll", "label": "roll"},
    {"id": "pitch", "label": "pitch"},
    {"id": "yaw", "label": "yaw"}],
  "cols": [{"id": "value", "unit": "mm, deg"}]}}
```

7.2.4.6 Datos de un grupo de estado

Los datos de un grupo de estado contienen datos TCP o del motor. Utilice «request_data» para recibir estos datos.

Clave	Valor	Tipo
state_group_data	El valor contiene el TCP del robot (id = «pose»), los datos del motor (id = «robot») o los estados de seguridad (id = «safety»).	Diccionario

```
{"state_group_data": {"id": "pose",
  "data": {"x": {"value": "0.0"},
    "y": {"value": "0.0"},
    "z": {"value": "1378.5"},
    "roll": {"value": "0.0"},
    "pitch": {"value": "90.0"},
    "yaw": {"value": "180.0"}}}}}
```

7.2.4.7 Abrir un cuadro de diálogo

Cuando se abre un cuadro de diálogo en la interfaz de usuario, se recibe el comando «open_dialog». A continuación, se puede responder al cuadro de diálogo con la acción «answer_dialog».

Clave	Valor	Tipo
open_dialog	El cuadro de diálogo adopta los siguientes valores: • id → ID único del diálogo (para responder al diálogo) • kind ○ 0 → cuadro de diálogo de confirmación (botones para confirmar/rechazar) ○ 1 → cuadro de diálogo de entrada (introducción de texto) ○ 2 → cuadro de diálogo de archivos (carga de archivos) ○ 3 → Diálogo desplegable (requiere una lista de opciones) ○ 4 → Cuadro de diálogo de selección (requiere una lista de opciones) • por defecto → Valor de retorno, en caso de que no se reciba respuesta • color ○ 0 → Información ○ 1 → Éxito ○ 2 → Advertencia ○ 3 → Error • título → título que se mostrará • text → Texto que se va a mostrar • options → Lista de opciones para cuadros de diálogo desplegables/de selección	Diccionario

```
{
  "open_dialog": {
    "id": "5cee85a8-9ab2-4de8-b1ea-52ded8cc68c0",
    "kind": 0, "default": true, "color": 2,
    "title": "Movement Warning",
    "text": "In order to Calibrate, the robot will move.
             Do you want to proceed?"
  }
}
```

7.2.4.8 Enviar desde la aplicación

Recibe un mensaje enviado desde una aplicación.

Clave	Valor	Tipo
script_send	Valor enviado por la aplicación	Varios

```
{
  "script_send": 123
}
```

7.2.5 Registro de mensajes: envío de comandos

Todos los comandos enviados y recibidos son cadenas [JSON](#) . Como se ha indicado anteriormente, los mensajes se codifican con el estándar **UTF-8** y se separan mediante el **carácter de fin de texto** 0x03 (en Python, «\x03»). Un comando enviado debe contener un diccionario codificado en JSON con tres claves obligatorias (y una opcional). Encontrará un ejemplo de código que envía mensajes al robot en la sección Enviar comandos.

action (función)	define qué comando se ejecuta	str
bridge (módulo)	define el módulo que contiene el comando, normalmente «core»	str
arguments (argumentos)	definen los argumentos del comando; consulte más abajo las descripciones detalladas de los argumentos	dict
sequence (identificador)	<opcional> Identificador del comando, que se devuelve al final en action_result_value	int

Tenga en cuenta que la siguiente lista no es exhaustiva, pero incluye los comandos más importantes.

7.2.5.1 Inicio de sesión con token

El primer comando que se debe enviar debe ser siempre una llamada de inicio de sesión para utilizar la conexión activa. Siga el procedimiento de inicio de sesión remoto descrito en la sección Autenticación segura para obtener un token de acceso válido.

Función	token_login	
Módulo	core	
Argumentos	token	session_id
Valor del argumento	el token de inicio de sesión personalizado proporcionado por el comando de inicio de sesión	el ID de sesión único proporcionado por el comando de inicio de sesión
Tipo de argumento	Cadena	Cadena

```
{"action": "token_login",
  "bridge": "core",
  "arguments": {"token": "MY_ACCESS_TOKEN",
               "session_id": "MY_SESSION_ID"}}
```

7.2.5.2 Solicitar el control activo

Obtener el control activo, de modo que la conexión TCP pase a ser la estación de control activa del robot. Tenga en cuenta que, al hacerlo, se envía una solicitud a la estación de control activa actual, que puede ser aceptada o rechazada.

Función	request_active_control
Módulo	core

```
{"action": "request_active_control",  
  "bridge": "core",  
  "arguments": {}}
```

7.2.5.3 Liberar el control activo

Ceder el control activo, de modo que la conexión TCP deje de ser la estación de control activa del robot.

Función	release_active_control
Módulo	core

```
{"action": "release_active_control",  
  "bridge": "core",  
  "arguments": {}}
```

7.2.5.4 Consultar datos

Este comando puede utilizarse como señal de vida. Recibe diferentes tipos de datos.

Función	request_data
Módulo	core

```
{"action": "request_data",  
  "bridge": "core",  
  "arguments": {}}
```

7.2.5.5 Puesta en marcha

Hace que el robot encienda sus motores.

Función	power_up
Módulo	core
Argumentos	tool_id
Valor del argumento	ID de una herramienta definida por el usuario o 0 para la detección automática de herramientas
Tipo de argumento	Entero o cadena

```
{  
  "action": "power_up",  
  "bridge": "core",  
  "arguments": {"tool_id": 0}  
}
```

7.2.5.6 Poner fuera de servicio

Interrumpe la conexión y hace que el robot apague sus motores.

Función	power_down
Módulo	core

```
{  
  "action": "power_down",  
  "bridge": "core",  
  "arguments": {}  
}
```

7.2.5.7 Pausar

Pausa todas las funciones de movimiento y aplicaciones en ejecución.

Función	pause
Módulo	core

```
{  
  "action": "pause",  
  "bridge": "core",  
  "arguments": {}  
}
```

7.2.5.8 Reanudar

Reanuda una aplicación o un movimiento, siempre que esté en pausa.

Función	resume
Módulo	core

```
{"action": "resume",  
  "bridge": "core",  
  "arguments": {}}
```

7.2.5.9 Activación del control manual (HGC)

Desbloquea las articulaciones del robot para que se encuentren en modo de compensación gravitacional y puedan moverse manualmente. El estado debe ser «Listo» y cambia a «Control manual».

Función	release
Módulo	core
Argumentos	joint_ids
Valor del argumento	<opcional> Lista de los ID de los ejes para los que se debe activar el modo manual. De forma predeterminada, el modo manual se activa para todos los ejes.
Tipo de argumento	Lista

```
{"action": "release",  
  "bridge": "core",  
  "arguments": {"joint_ids": ["1", "2", "5", "6"]}}
```

7.2.5.10 Desactivación del control manual (HGC)

Mantiene las articulaciones del robot de modo que se encuentren en modo de control de posición y ya no puedan moverse manualmente. El estado debe ser «Control manual» y cambia a «Listo».

Función	hold
Módulo	core
Argumentos	joint_ids
Valor del argumento	<opcional> Lista de los ID de los ejes que se deben mantener. De forma predeterminada, se mantienen todos los ejes.
Tipo de argumento	Lista

```
{"action": "hold",  
  "bridge": "core",  
  "arguments": {"joint_ids": ["1", "2", "5", "6"]}}
```

7.2.5.11 Detener

Interrumpe todas las funciones de movimiento y aplicaciones en ejecución.

Función	stop
Módulo	core

```
{"action": "stop",  
  "bridge": "core",  
  "arguments": {}}
```

7.2.5.12 Cambiar de proyecto

Cambia el proyecto de base de datos actual al proyecto con un ID determinado. El estado debe ser «Listo».

Función	change_project
Módulo	core
Argumentos	project_id
Valor del argumento	ID del proyecto al que se desea cambiar. El proyecto predeterminado tiene el ID 1.
Tipo de argumento	Entero

```
{"action": "change_project",  
  "bridge": "core",  
  "arguments": {"project_id": 1}}
```

7.2.5.13 Ejecutar una aplicación

Ejecuta una aplicación especificada por su nombre o su ID. El estado debe ser «Listo» y cambia a «En ejecución». Mientras se ejecuta una aplicación, el robot puede ponerse en pausa. Tan pronto como finalice, el estado vuelve a «Listo».

Función	execute_script	
Módulo	core	
Argumentos	script_name	script_id
valor del argumento	El nombre de la aplicación que se va a ejecutar.	El ID de la aplicación que se va a ejecutar.
Tipo de argumento	Cadena	Entero

```
{"action": "execute_script",  
  "bridge": "core",  
  "arguments": {"script_id": 3}}
```

```
{"action": "execute_script",  
  "bridge": "core",  
  "arguments": {"script_name": "my_application"}}
```

7.2.5.14 Probar la aplicación

Ejecuta el código de una aplicación en modo de prueba. El estado debe ser «Listo» y cambia a «En ejecución». Mientras una aplicación está en ejecución, se puede pausar el robot. Tan pronto como finaliza, el estado vuelve a «Listo».

Función	test_script		
Módulo	core		
Argumentos	código de script	tipo_de_script	script_id
Valor del argumento	El código Python de la aplicación que se va a probar.	El tipo de script: • «main» • «parallel» • «background»	El ID del script que se va a probar. Si el script no existe en la base de datos, el ID es 0.
Tipo de argumento	Cadena	Cadena	Entero

```
{
  "action": "test_script",
  "bridge": "core",
  "arguments": {
    "script_code": "print('Hello World')",
    "script_type": "main",
    "script_id": 0
  }
}
```

7.2.5.15 Recibir un mensaje

Utilice esta función para enviar un mensaje desde su canal de conexión a una aplicación. El mensaje se puede recibir en la aplicación mediante la función **receive_message**.

Función	script_receive
Módulo	core
Argumentos	mensaje
Valor del argumento	Mensaje recibido por la aplicación.
Tipo de argumento	Varios

```
{
  "action": "script_receive",
  "bridge": "core",
  "arguments": {
    "message": 123
  }
}
```

7.2.5.16 Desplazarse a una pose

Mueve al robot a una pose con un ID o un nombre específicos. El estado debe ser «Listo» y cambia a «En ejecución». Mientras el robot se mueve, el movimiento se puede pausar. Una vez finalizado el movimiento, el estado vuelve a «Listo».

Función	move_pose
Módulo	core
Argumentos	pose_id

Valor del argumento	ID o nombre de la pose a la que se debe dirigir.
Tipo de argumento	Entero o cadena

```
{
  "action": "move_pose",
  "bridge": "core",
  "arguments": {
    "pose_id": 2
  }
}
```

```
{
  "action": "move_pose",
  "bridge": "core",
  "arguments": {
    "pose_id": "start_pose"
  }
}
```

7.2.5.17 Reproducir ruta

Reproduce la ruta con un ID o un nombre específico. El estado debe ser «Listo» y cambia a «En ejecución». Mientras el robot se mueve, el movimiento se puede pausar. Una vez finalizado el movimiento, el estado vuelve a «Listo».

Función	play_path
Módulo	core
Argumentos	path_id
Valor del argumento	ID o nombre de la ruta que se va a ejecutar.
Tipo de argumento	Entero o cadena

```
{"action": "play_path",  
  "bridge": "core",  
  "arguments": {"path_id": 2}}
```

```
{"action": "play_path",  
  "bridge": "core",  
  "arguments": {"path_id": "p1"}}
```

7.2.5.18 Cambiar herramienta

Cambia la función de la herramienta, normalmente «Desplazar» y «Colocar». El estado debe ser «Listo» y cambia a «En ejecución». Una vez completada la función, el estado vuelve a «Listo».

Función	toggle_tool
Módulo	core
Argumentos	pick
Valor del argumento	true para levantar un objeto, false para colocar un objeto. Esto viene determinado por el comportamiento de agarre.
Tipo de argumento	Booleano

```
{"action": "toggle_tool",  
  "bridge": "core",  
  "arguments": {"pick": false}}
```

7.2.5.19 Escribir salidas digitales y analógicas

Describe una o varias salidas (digitales o analógicas) del controlador principal o del controlador de la herramienta (líneas de señal que se transmiten desde el controlador principal a la herramienta). Para cada una de estas combinaciones se utiliza una función propia (con los mismos argumentos). Encontrará una lista de todas las E/S en el menú de configuración de E/S o en el menú de E/S.

Función	write_digital_main_output write_analog_main_output write_digital_tool_output write_analog_tool_io	
Módulo	core	
Argumentos	identificadores	valores
valor del argumento	Existen varias formas de identificar una E/S concreta, ya sea mediante • número (entero) • nombre definido por el usuario (cadena) Si se escriben varias E/S al mismo tiempo, puede combinar números o nombres definidos por el usuario en una lista.	Dependiendo del tipo de «identificadores» seleccionado, los valores pueden ser: • verdadero/falso (booleano) para E/S digitales • números (flotante) para E/S analógicas o una lista de estos valores, si se escriben varias E/S al mismo tiempo.
Tipo de argumento	Entero / Cadena / Lista	Booleano / Flotante / Lista

```
{
  "action": "write_digital_tool_io",
  "bridge": "core",
  "arguments": {
    "identifiers": 3,
    "values": true
  }
}
```

Este ejemplo describe la salida digital de herramienta 3 y la establece en «true».

```
{
  "action": "write_analog_tool_io",
  "bridge": "core",
  "arguments": {
    "identifiers": "my_analog_output",
    "values": 9.5
  }
}
```

Este ejemplo describe la salida analógica de la herramienta, a la que se le ha asignado el nombre definido por el usuario «my_analog_output», y la establece en 9,5 V.

```
{
  "action": "write_digital_main_io",
  "bridge": "core",
  "arguments": {
    "identifiers": [1, 2],
    "values": [true, false]
  }
}
```

Este ejemplo describe las salidas digitales principales 1 y 2 y establece 1 en true y 2 en false.

7.2.5.20 Responder a un cuadro de diálogo

Responde a una ventana de diálogo abierta. La respuesta depende del tipo de diálogo.

Función	answer_dialog	
Módulo	core	
Argumentos	dialog_id	dialog_answer
Valor del argumento	El identificador único del diálogo al que se envía la respuesta. El ID forma parte de la respuesta « open_dialog » recibida.	La respuesta que se debe enviar a este diálogo.
Tipo de argumento	Cadena	Booleano / Entero / Cadena

```
{
  "action": "answer_dialog",
  "bridge": "core",
  "arguments": {
    "dialog_id": "5cee85a8-9ab2-4de8-b1ea-52ded8cc68c0",
    "dialog_answer": true
  }
}
```

7.2.5.21 Establecer variable de estado

Establece el valor de una variable de estado «variable_name» en «variable_value». La variable debe existir ya en la aplicación con el ID «script_id». Además, la aplicación debe estar almacenada en la base de datos. Este comando se puede ejecutar mientras se está ejecutando una aplicación.

Función	set_state_variable		
Módulo	core		
Argumentos	script_id	nombre_de_la_variabl e	valor_de_la_varia ble
valor del argumento	ID de la aplicación que contiene la variable de estado que se va a modificar.	Nombre de la variable de estado que se va a modificar.	Nuevo valor.
Tipo de argumento	Entero	Cadena	Varios

```
{
  "action": "set_state_variable",
  "bridge": "core",
  "arguments": {
    "script_id": 2,
    "variable_name": "integer_1",
    "variable_value": 42
  }
}
```

7.2.5.22 Establecer una variable global

Establece el valor de una variable global «name» en «value». Las variables globales se almacenan más allá del tiempo de ejecución de una aplicación y se conservan incluso tras reiniciar el robot.

Función	set_global_variable	
Módulo	core	
Argumentos	nombre	valor
valor del argumento	Nombre de la variable global	Valor de la variable global
Tipo de argumento	Cadena	Varios

```
{"action": "set_global_variable",  
  "bridge": "core",  
  "arguments": {"name": "my_variable", "value": 3}}
```

7.2.5.23 Leer variable global

Lee el valor de una variable global «name». Las variables globales se almacenan más allá del tiempo de ejecución de una aplicación y se conservan incluso después de reiniciar el robot.

Función	get_global_variable	
Módulo	core	
Argumentos	nombre	
valor del argumento	Nombre de la variable global	
Tipo de argumento	Cadena	

```
{"action": "get_global_variable",  
  "bridge": "core",  
  "arguments": {"name": "my_variable"}}
```

7.2.5.24 Establecer una variable compartida

Establece el valor de una variable compartida «name» en «value». Las variables compartidas solo están disponibles durante el tiempo de ejecución de una aplicación. Se eliminan una vez que la aplicación ha finalizado o se ha detenido el robot.

Función	set_shared_variable	
Módulo	core	
Argumentos	nombre	valor
valor del argumento	Nombre de la variable compartida	Valor de la variable compartida
Tipo de argumento	Cadena	Varios

```
{"action": "set_shared_variable",  
  "bridge": "core",  
  "arguments": {"name": "my_variable", "value": 3}}
```

7.2.5.25 Leer variable compartida

Lee el valor de una variable compartida «name». Las variables compartidas solo están disponibles durante el tiempo de ejecución de una aplicación. Se eliminan una vez que la aplicación ha finalizado o se ha detenido el robot.

Función	get_shared_variable	
Módulo	core	
Argumentos	nombre	
Valor del argumento	Nombre de la variable compartida	
Tipo de argumento	Cadena	

```
{"action": "get_shared_variable",  
  "bridge": "core",  
  "arguments": {"name": "my_variable"}}
```

7.3 Módulo de conexión REST

7.3.1 Establecer conexión

El módulo de conexión REST proporciona el recurso **https://<DIRECCIÓN_IP>/rest/gateway** para recibir comandos de los clientes conectados. Tenga en cuenta que solo hay un único canal de conexión. Si varios usuarios se conectan a través de REST, comparten este canal.

Utilice el siguiente script de ejemplo para tomar el control del robot a través de la conexión REST y, a continuación, enviar comandos al robot. Tenga en cuenta que, antes de enviar comandos, debe haber iniciado sesión y tener el control activo. Por lo tanto, se recomienda compartir el control y cerrar la sesión tan pronto como la conexión ya no sea necesaria.

```
import json
import requests

IP_ADDRESS = "10.0.0.203"
TOKEN = "MY_ACCESS_TOKEN"
SESSION_ID = "MY_SESSION_ID"

REST_URL = f"https://{IP_ADDRESS}/rest/gateway"
HEADERS = {"Accept": "application/json", "Content-Type": "application/json"}

def request(method, url, headers, data, timeout=None):
    try:
        response = method(url=url, headers=headers, data=data, timeout=timeout,
verify=False)
        response_code = response.status_code

        # this is equal to 'response.status_code < 400' (errors have codes 400+)
        if response.ok:
            # try to parse json payload
            try:
                response_payload = response.json()
            except json.decoder.JSONDecodeError:
                # usually this happens because the response payload is empty
                # this happens for example when a request returns "", 200
                response_payload = dict()
                print("response payload json was empty")
            else:
                print("response.ok returned False")

        except requests.Timeout:
            response_payload = dict()
            response_code = 408 # timeout
        except Exception as e:
            # usually connection problems...
            print(f"rest client could not connect to specified endpoint: {e}")
            response_payload = dict()
            response_code = 404 # not found

        return response_code, response_payload

def post_request(data, timeout=None):
    return request(method=requests.post, url=REST_URL, headers=HEADERS, data=data,
```

```
timeout=timeout)

# login
data = {"bridge": "core", "action": "token_login", "arguments": {"token": "<TOKEN>",
"session_id": "<SESSION_ID>"}}
response = post_request(data, timeout=10.0)
# acquire the active control before start sending commands to the robot
data = {"bridge": "core", "action": "request_active_control", "arguments": {}}
response = post_request(data, timeout=1.0)

# add your commands here

# when you're done using this connection, give away the active control
data = {"bridge": "core", "action": "release_active_control", "arguments": {}}
response = post_request(data, timeout=1.0)
# when you're done working with this connection, you should logout
data = {"bridge": "core", "action": "logout", "arguments": {}}
response = post_request(data, timeout=1.0)
```

7.3.2 Envío de comandos

Además de los comandos mencionados en el ejemplo anterior, hay muchos otros disponibles. Consulte la lista de comandos del módulo de conexión TCP, ya que se utilizan los mismos comandos para la conexión REST.

Además de enviar un parámetro «bridge», «action», «arguments» y un parámetro opcional «sequence», también puede enviar un parámetro opcional «timeout». Si envía un tiempo de espera, la solicitud REST no devuelve la respuesta estándar (200, {}) si se envía correctamente, sino que el comando se ejecuta de forma bloqueante, y recibirá la respuesta como (200, action_return_value), donde el valor de retorno de la acción contiene la información sobre qué función de la API se ha invocado, si la invocación se ha realizado con éxito y qué valor de retorno ha devuelto la función invocada.

7.3.3 Configuración

En el menú de configuración de la interfaz de escritorio encontrará una opción para desactivar la autenticación del módulo de conexión REST. Al desactivar la autenticación, ya no es necesario iniciar ni cerrar sesión.

Authentication Settings

Disable Authentication for REST



Figura 7.3.3: Ajustes del módulo REST

7.4 Módulo de conexión ROS

El módulo de conexión ROS le permite integrar su robot en un entorno ROS 2 (Robot Operating System) y comunicarse con AX Portal mediante comandos ROS 2. Las instrucciones de este capítulo son compatibles con ROS Jazzy y [CycloneDDS](#), así como con [ROS-Middleware \(RMW\)](#).

 ADVERTENCIA	El módulo ROS ofrece acceso a determinadas funciones de bajo nivel. Por ejemplo, se pueden enviar comandos de control directos a los motores del robot, que se ejecutan de forma inmediata. En la interfaz de usuario, algunas funciones requieren comprobaciones adicionales o confirmaciones mediante cuadros de diálogo que pueden no estar disponibles aquí. Por lo tanto, tenga especial cuidado al controlar su robot directamente a través de ROS.
---	--

Nota:

- Esta es la primera versión de AX Portal con ROS 2; se esperan funciones adicionales en el futuro.
- Actualmente, el módulo ROS no está disponible con el simulador de AX Portal, sino exclusivamente con robots reales.

7.4.1 Instalación

Para comunicarse con el módulo ROS de AX Portal, necesita tener ROS instalado en su(s) ordenador(es). Para la configuración, la [documentación oficial de ROS](#) ofrece diversas opciones, como el paquete deb, la imagen de Docker, la instalación desde el código fuente, etc. En este ejemplo, utilizamos la imagen oficial de Docker de ROS para comunicarnos con el robot. El siguiente archivo docker-compose crea un entorno ROS sencillo en su ordenador:

```
services:
  ros2_development:
    image: osrf/ros:jazzy-desktop
    container_name: "ros2_development"
    command: /bin/bash -c
      "apt update &&
      apt install -y ros-jazzy-rmw-cyclonedds-cpp &&
      sleep infinity"
    environment:
      - RMW_IMPLEMENTATION=rmw_cyclonedds_cpp
      - CYCLONEDDS_URI=/ros/cyclonedds_config.xml

    network_mode: "host"
```

```
volumes:  
- "./cyclonedds_config.xml:/ros/cyclonedds_config.xml"
```

Tenga en cuenta que, al iniciar, instalamos `ros-jazzy-rmw-cyclonedds-cpp` y establecemos `RMW_IMPLEMENTATION` en `rmw_cyclonedds_cpp` para indicar a ROS que utilice CycloneDDS, el mismo RMW que se ejecuta en el robot. Además, compartimos un archivo `cyclonedds_config.xml` y establecemos `CYCLONEDDS_URI` en su ubicación para utilizar una configuración personalizada de CycloneDDS.

7.4.2 Configuración

La comunicación ROS entre el robot y su ordenador requiere ajustes en ambos dispositivos. En su ordenador portátil, debe establecer una nueva variable de entorno y proporcionar un archivo de configuración de red. En el robot no tiene acceso directo a estos elementos, pero puede editarlos en la interfaz de usuario, en la sección «ROS» del menú de configuración:

The image shows a user interface titled "ROS Function Settings". It contains several configuration options:

- Network Interface:** A dropdown menu currently showing "Wifi".
- Multicast:** A toggle switch that is currently turned off (indicated by a grey circle with an 'X').
- IP Addresses of ROS 2 Peers:** A list of input fields. The first two fields contain "192.168.0.1" and "192.168.0.2". Each field has a trash icon to its right. There is a plus icon at the bottom right of the list to add more addresses.
- ROS Domain ID:** A dropdown menu currently showing "15".

Figura 7.4.2: Ajustes del módulo ROS

En las siguientes secciones aprenderá el significado de cada uno de los ajustes. Además, descubrirá qué ajustes correspondientes son necesarios en su ordenador.

7.4.3 Detección de red con Cyclone DDS

Ya se ha mencionado Cyclone DDS en numerosas ocasiones. ¿En qué consiste? Como usuario de AX Portal, debe saber lo siguiente:

Cuando se selecciona Cyclone DDS como middleware de ROS, este se encarga de distribuir los paquetes de datos relacionados con ROS en su red. Es responsable de establecer una conexión de red entre sus dispositivos para que ROS 2 pueda enviar datos a través de dichas conexiones.

7.4.3.1 Interfaz de red

En primer lugar, Cyclone DDS debe saber qué interfaz de red utiliza ROS. La configuración predeterminada es Ethernet.

Utilice la configuración «Interfaz de red» en «Configuración de funciones ROS» para seleccionar la interfaz deseada.

Nota: Al iniciar el módulo ROS a través de AX Portal, Cyclone DDS escanea la interfaz de red en busca de direcciones IP de equipos pares. Si no se puede acceder a esta interfaz, el módulo ROS no se carga. Por lo tanto, si ha configurado la interfaz de red en Ethernet, asegúrese de que el cable esté conectado antes de iniciar el robot.

7.4.3.2 Métodos de detección

Cyclone DDS ofrece dos métodos para detectar otros dispositivos en la red: multidifusión y punto a punto. En el caso de la multidifusión, cada paquete de red se envía a todos los clientes accesibles en el ámbito de la red. La ventaja radica en su simplicidad: no es necesario especificar qué equipos pueden comunicarse entre sí. En esta configuración, la variable de entorno ROS_DOMAIN_ID (véase más abajo) sigue estando disponible para evitar que dos robots mezclen mensajes.

La desventaja es la sobrecarga de la red: todos los paquetes son procesados por cada cliente de la red. Ni siquiera el establecimiento de diferentes ROS_DOMAIN_IDs puede evitarlo. Incluso si en un ordenador no está instalado ROS: dado que Cyclone DDS funciona sobre la base de UDP, todos los ordenadores de la red reciben y procesan estos paquetes UDP de todos modos. Para redes más pequeñas y cantidades reducidas de datos, esto no suele suponer un problema. Sin embargo, si, por ejemplo, un robot enviase una transmisión de cámara a todos los clientes de una red grande, esto podría ralentizar notablemente toda la red. En un escenario de este tipo, es más seguro especificar explícitamente en la configuración de Cyclone DDS un conjunto de direcciones IP que puedan comunicarse entre sí a través de conexiones peer-to-peer.

En las dos subsecciones siguientes se explica cómo configurar su red para la comunicación multicast o peer-to-peer. Nota: En la mayoría de las distribuciones de Linux, el comando `ip addr show` resulta útil para determinar sus direcciones IP y los nombres de las interfaces de red correspondientes.

7.4.3.2.1 Peer-to-Peer

En las instrucciones de instalación anteriores, hemos incluido un archivo de configuración de Cyclone DDS en el archivo docker-compose. A continuación se muestra una configuración de ejemplo para Cyclone DDS, con el fin de configurar su equipo para la comunicación entre pares:

```
<?xml version='1.0' encoding='utf-8'?>
<CycloneDDS xmlns="https://cdds.io/config"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="https://cdds.io/config
https://raw.githubusercontent.com/eclipse-cyclonedds/cyclonedds/master/etc/cyclonedds.xsd"
">
  <Domain id="any">
    <General>
      <Interfaces>
        <NetworkInterface name="INTERFACE_NAME" multicast="false" />
      </Interfaces>
      <AllowMulticast>false</AllowMulticast>
      <EnableMulticastLoopback>false</EnableMulticastLoopback>
    </General>
    <Discovery>
```

```
<MaxAutoParticipantIndex>119</MaxAutoParticipantIndex>
<ParticipantIndex>auto</ParticipantIndex>
<Peers>
  <Peer address="192.168.1.1" />
  <Peer address="192.168.1.2" />
</Peers>
</Discovery>
</Domain>
</CycloneDDS>
```

En primer lugar, sustituya `INTERFACE_NAME` por el nombre real de la interfaz de red de su ordenador que se utiliza para la comunicación ROS. Ejemplos: `enp1s0`, `wlp2s0`, etc.

A continuación, adapte la lista que aparece en «Peers» para que refleje las direcciones IP de los ordenadores de su red que desea conectar entre sí a través de ROS.

De forma análoga al archivo anterior, en su robot hay almacenado un archivo similar. No puede acceder a él directamente, pero puede editar su contenido a través del AX Portal, en la sección «Configuración de funciones ROS» del menú de configuración. Allí, el ajuste «Direcciones IP de pares ROS 2» ofrece una lista de campos de texto que puede rellenar con direcciones IP.

Tenga en cuenta lo siguiente:

- Es necesario reiniciar el robot para que el cambio surta efecto.
- Si una dirección IP de la lista es incorrecta, AX Portal mostrará un error y la nueva lista no se guardará.
- La lista debe incluir la propia dirección IP del robot. Actualmente, esta no se lee automáticamente.
- Si se elimina una dirección IP de la lista y se guarda la nueva lista, la dirección IP también se eliminará de la configuración de Cyclone DDS.
- Debido a la configuración del cortafuegos, puede ser necesario que todos los participantes de ROS se encuentren en la misma subred.

7.4.3.2.2 Multidifusión

A continuación se muestra otra configuración de ejemplo para Cyclone DDS con el fin de configurar su ordenador para la multidifusión:

```
<?xml version="1.0" encoding="utf-8"?>
<CycloneDDS
  xmlns="https://cdds.io/config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="https://cdds.io/config
https://raw.githubusercontent.com/eclipse-cyclonedds/cyclonedds/master/etc/cyclonedds.xsd"
>
  <Domain Id="any">
    <General>
      <Interfaces>
```

```
<NetworkInterface name="INTERFACE_NAME" multicast="true" />
</Interfaces>
<AllowMulticast>true</AllowMulticast>
<EnableMulticastLoopback>true</EnableMulticastLoopback>
</General>
</Domain>
</CycloneDDS>
```

Sustituya también aquí `INTERFACE_NAME` por el nombre real de la interfaz de red de su ordenador que se utiliza para la comunicación ROS. Ejemplos: `enp1s0`, `wlp2s0`, etc.

Para configurar su robot en modo multicast, simplemente active el botón de configuración «Multicast». También en este caso es necesario reiniciar el robot para que el cambio surta efecto.

7.4.4 Establecer la conexión

Para iniciar ahora el entorno ROS en su ordenador, vaya a su archivo `docker-compose` y ejecute:

```
docker compose up -d
```

Compruebe si el contenedor Docker está en ejecución con

```
docker ps -a
```

lo que generará un resultado similar al siguiente:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
132bdd6fa4de	osrf/ros:jazzy-desktop	"/ros_..."	...	...	...	...

Cambie al modo interactivo del contenedor con el siguiente comando:

```
docker exec -it ros2_development bash
```

No olvide cargar el archivo de configuración de ROS cada vez que acceda al contenedor:

```
source /opt/ros/jazzy/setup.bash
```

Para poder operar varios robots en la misma red, utilizamos un `ROS_DOMAIN_ID` diferente en cada robot. En AX Portal, abra «Configuración de funciones ROS» y establezca el ROS Domain ID deseado. El rango permitido es de 6 a 41. Tenga en cuenta también que, tras esta acción, es necesario reiniciar el robot. Vuelva ahora al terminal en el que tiene abierto su contenedor Docker de ROS y establezca el mismo valor para `ROS_DOMAIN_ID`:

```
export ROS_DOMAIN_ID=SELECTED_NUMBER
```

Ahora debería poder ver un nodo AX Portal-ROS llamado «robot» con el siguiente comando:

```
ros2 node list
```

7.4.5 Acceso a los datos del robot a través de ROS

En cuanto vea el nodo, podrá visualizar la lista de temas disponibles ejecutando lo siguiente:

```
ros2 topic list
```

A través de los temas de ROS 2, puede transmitir datos en tiempo real desde su robot. Por ejemplo, para suscribirse a las posiciones de los ejes, velocidades, etc., ejecute lo siguiente:

```
ros2 topic echo /joint_states
```

7.4.6 Funcionalidades de ROS 2 en scripts

Puede importar rclpy directamente en scripts. Esto le permite crear objetos de ROS 2 como editores, suscriptores, etc.

El módulo ROS crea su propio nodo ROS. Utilice la función `ros_handler.ros_node()` para acceder a él. Atención: No ejecute `rclpy.spin` en este nodo, ya que esto ya se realiza dentro del módulo ROS.

7.4.7 Llamar a una acción

El nodo ROS de AX Portal crea un servidor de acciones denominado «/robot/execute_function» para ejecutar funciones de puente. La acción tiene una definición personalizada `ExecuteFunction` del paquete `generic_msgs` (<https://github.com/Epson-Robots/epson-ax-ros2>), que debe instalarse para obtener la definición de la acción. Clone el paquete en su espacio de trabajo y compílelo con `colcon`:

```
mkdir -p /colcon_ws/src
cd /colcon_ws/src
git clone https://github.com/Epson-Robots/epson-ax-ros2
cd ..
colcon build --packages-up-to generic_msgs
source install/setup.bash
```

Para enviar acciones ROS2 al robot, su cliente debe asumir el control activo. Para ello, ejecute el siguiente comando en su terminal y confirme el cuadro de diálogo en la interfaz gráfica de usuario del portal AX:

```
ros2 action send_goal --feedback /execute_function generic_msgs/action/ExecuteFunction  
"{action: request_active_control, bridge: core, arguments: '{}'}"
```

Ahora puede ejecutar una función de puente desde un terminal con los siguientes comandos:

```
ros2 action send_goal --feedback /execute_function generic_msgs/action/ExecuteFunction  
"{action: read_actuator_position, bridge: core, arguments: '{\"actuator_ids\": 6}'}"  
  
ros2 action send_goal /execute_function generic_msgs/action/ExecuteFunction "{action:  
move_joints, bridge: core, arguments: '{\"joints\": 6, \"joint_position\": 100.0}'}"
```

7.4.8 Configuración de autenticación

Actualmente, el módulo ROS solo funciona sin autenticación. En el menú de configuración de la interfaz de escritorio, debe desactivar manualmente la autenticación para este módulo de conexión específico.

Authentication Settings

Disable Authentication for ROS



Figura 7.4.8: Ajustes de autenticación de ROS

7.5 Módulo de conexión Modbus

7.5.1 Resumen

El protocolo utilizado para el módulo Modbus es [TCP Modbus](#) . Por parte del servidor, el módulo se basa en Python y se ha implementado utilizando la [biblioteca PyModbus](#). Encontrará más detalles en la [documentación](#) de PyModbus. El módulo se divide en registros de lectura (registros de entrada analógica, 30000-39999) y registros de escritura (registros de retención de salida analógica, 40000-49999). Para comunicarse con el servidor se puede utilizar cualquier cliente TCP Modbus. El puerto que debe utilizarse es el 5020. En la última sección de este capítulo encontrará varios ejemplos de código Python, algunos de los cuales explican cómo establecer una conexión con el servidor mediante un script de Python.

7.5.2 Configuración

Actualmente, la conexión Modbus solo funciona sin autenticación. En el menú de configuración de la interfaz de escritorio, debe desactivar manualmente la autenticación para este módulo de conexión específico.

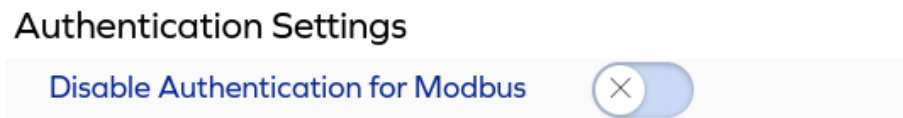


Figura 7.5.2: Ajustes del módulo Modbus

7.5.3 Registros de escritura (descripción sencilla)

El módulo Modbus responde a cualquier cambio de información tanto en el registro de comandos como en el registro de propiedades. El borrado de los datos de los registros de escritura es responsabilidad del usuario.

7.5.3.1 Registro de comandos

Escriba un solo bit en el registro de comandos (30000) para ejecutar un comando. Algunos comandos requieren un argumento en el registro de datos de comando (30001). Si no puede escribir en ambos registros al mismo tiempo, asegúrese de escribir primero en el registro de datos de comando.

Dirección absoluta: 30000 - 30001 Dirección relativa: 0x000 - 0x001		
Bit	Registro de instrucciones (30000)	Datos (30001)
0	Puesta en marcha	ID de herramienta
1	Poner fuera de servicio	
2	Detener	
3	Pausa	
4	Reanudar	
5	Calibrar	ID de la aplicación
6	Activar control manual (HGC)	
7	Desactivar el control manual (HGC)	
8	Cambiar de proyecto	ID del proyecto
9	Ejecutar aplicación	ID de la aplicación
10	Ir a la pose	ID de la pose
11	Reproducir ruta	ID de ruta
12	Colocar herramienta	
13	Retirar herramienta	
14 ... 15	<i>Reservado</i>	

7.5.3.2 Registro de propiedades

Escriba un solo bit en el registro de propiedades (30002) para ejecutar un comando. Algunos comandos requieren un argumento en el registro de datos de propiedades (30003). Si no puede escribir en ambos registros al mismo tiempo, asegúrese de escribir primero en el registro de datos de propiedades.

Dirección absoluta: 30002 - 30003 Dirección relativa: 0x002 - 0x003		
Bit	Registro de propiedades (30002)	Datos (30003)
0	Escribir salidas digitales	Estado de E/S
1	Establecer eventos de aplicación	ID del evento
2	Confirmar error	
3	Leer posición	
4	Solicitar control activo	
5	Habilitar el control activo	
6	Cambiar al modo manual	
7	Cambiar al modo automático	
8	Restablecer la función de seguridad	
9	Reanudar el movimiento	
10 ... 15	<i>reservado</i>	

7.5.3.3 Registros de escritura definidos por el usuario

Dirección absoluta: 30016 Dirección relativa: 0x010		...	Dirección absoluta: 30255 Dirección relativa: 0x0FF	
0 ... 15	Datos definidos por el usuario		0 ... 15	Datos definidos por el usuario

Utilice los registros de escritura de usuario para enviar información a su aplicación. Con la función **modbus.write_user_register** de la aplicación se pueden leer datos definidos por el usuario.

7.5.4 Registros de escritura (descripción detallada)

7.5.4.1 No realizar ninguna acción

El borrado de los datos de los registros de escritura es responsabilidad del usuario. Para borrar el registro de comandos (30000) o el registro de propiedades (30002), simplemente escriba el valor entero 0.

0x000	0x001	0x002	0x003
0		0	

7.5.4.2 Puesta en marcha

Antes de comenzar a trabajar con el robot, este debe encender sus motores. El ID de la herramienta definida por el usuario que se va a conectar se indica mediante el número n. El robot cambia su estado de «Fuera de servicio» (1, bit 0) a «Listo» (8, bit 3).

0x000	0x001	0x002	0x003
1	n		

7.5.4.3 Poner fuera de servicio

Una vez finalizada la jornada laboral, el robot puede apagar sus motores. El robot puede apagarse en cualquier estado. Tan pronto como el robot se apaga, el estado cambia a **Fuera de servicio** (1, bit 0).

0x000	0x001	0x002	0x003
2			

7.5.4.4 Parada

Interrumpa todos los movimientos del robot y cambie el estado de «En marcha» (64, bit 6) a «Listo» (8, bit 3).

0x000	0x001	0x002	0x003
4			

7.5.4.5 Pausa

Detenga el robot mientras se encuentra en el estado «En marcha» (64, bit 6). El estado cambia a «En pausa» (128, bit 7).

0x000	0x001	0x002	0x003
8			

7.5.4.6 Continuar

Reanude el robot si se encuentra en estado **Pausado** (128, bit 7). El estado cambia a **En marcha** (64, bit 6).

0x000	0x001	0x002	0x003
16			

7.5.4.7 Calibración

La calibración no tiene ningún efecto en este momento. Este bit está reservado.

0x000	0x001	0x002	0x003
32			

7.5.4.8 Activar el control manual (HGC)

Active el control manual para todas las articulaciones del robot, de modo que estas se encuentren en modo de compensación de gravedad y puedan moverse manualmente. El estado debe ser «**Listo**» (8, bit 3) y cambia a «**Control manual**» (2048, bit 11).

0x000	0x001	0x002	0x003
64			

7.5.4.9 Desactivar el modo de control manual (HGC)

Mantenga todas las articulaciones del robot de modo que se encuentren en el modo de control de posición y ya no puedan moverse manualmente. El estado debe ser «**Control manual**» (2048, bit 11) y cambia a «**Listo**» (8, bit 3).

0x000	0x001	0x002	0x003
128			

7.5.4.10 Cambiar de proyecto

Cambie el proyecto de base de datos actual al proyecto con el ID n. El estado debe ser **Listo** (8, bit 3).

0x000	0x001	0x002	0x003
256	n		

7.5.4.11 Iniciar la aplicación

Ejecute la aplicación con el ID n. El estado debe ser «**Listo**» (8, bit 3) y cambia a «**En ejecución**» (64, bit 6). Mientras se ejecuta una aplicación, se puede poner el robot en pausa. Tan pronto como finalice, el estado vuelve a «**Listo**» (8, bit 3).

0x000	0x001	0x002	0x003
512	n		

7.5.4.12 Desplazarse a la pose

Mueva el robot a la pose con el ID n. El estado debe ser **Listo** (8, bit 3) y cambia a **En ejecución** (64, bit 6). Mientras el robot se mueve, el movimiento puede pausarse. Tan pronto como finalice el movimiento, el estado vuelve a **Listo** (8, bit 3).

0x000	0x001	0x002	0x003
1024	n		

7.5.4.13 Reproducir ruta

Reproduzca la ruta con el ID n. El estado debe ser «**Listo**» (8, bit 3) y cambia a «**En ejecución**» (64, bit 6). Mientras el robot se desplaza, el movimiento puede pausarse. Tan pronto como finalice el movimiento, el estado vuelve a «**Listo**» (8, bit 3).

0x000	0x001	0x002	0x003
2048	n		

7.5.4.14 Colocar herramienta

Coloque un objeto con ayuda de la herramienta (por ejemplo, abriendo la pinza). El estado debe ser «**Listo**» (8, bit 3) y cambia a «**En ejecución**» (64, bit 6). Mientras la pinza se mueve, el movimiento puede pausarse. Tan pronto como finaliza el movimiento, el estado vuelve a «**Listo**» (8, bit 3).

0x000	0x001	0x002	0x003
4096			

7.5.4.15 Levantar la herramienta

Levante un objeto con la ayuda de la herramienta (por ejemplo, cerrando la pinza). El estado debe ser «**Listo**» (8, bit 3) y cambia a «**En ejecución**» (64, bit 6). Mientras la pinza se mueve, el movimiento puede pausarse. Tan pronto como finaliza el movimiento, el estado vuelve a «**Listo**» (8, bit 3).

0x000	0x001	0x002	0x003
8192			

7.5.4.16 Escritura en salidas digitales

Función reservada para la escritura de salidas digitales.

0x000	0x001	0x002	0x003
		1	d

7.5.4.17 Configuración de eventos de aplicación

Si una aplicación de usuario debe esperar un evento enviado por el dispositivo Modbus, se pueden utilizar los eventos de aplicación. Hay un total de 16 eventos disponibles. Dependiendo del valor de datos d, los eventos se activan o se borran. La función **de aplicación modbus.wait_for_event** espera hasta que se active un evento determinado. Si d es igual a 0, se borran todos los eventos. Si d es igual a 65535, se activan todos los eventos. Si d es igual a 9, se activan el primer y el cuarto evento, y se borran los restantes.

0x000	0x001	0x002	0x003
		2	d

7.5.4.18 Confirmar errores

Los errores deben confirmarse antes de ser borrados. Sin embargo, esto solo es necesario para leer los errores. La confirmación no afecta al estado del robot. Si tanto el código de error mayor como el menor (40004-40005) son iguales a 0, no es necesario confirmar ningún otro error.

0x000	0x001	0x002	0x003
		4	

7.5.4.19 Lectura de la posición

Actualice los datos de posición del robot. Los ángulos de articulación y las coordenadas del robot actuales se escriben en los registros de lectura correspondientes (40256-40267) tan pronto como se solicitan los datos de posición.

0x000	0x001	0x002	0x003
		8	

7.5.4.20 Solicitar control activo

Solicite el control activo para poder enviar comandos al robot. Se puede acceder a los registros de lectura de datos sin control activo. Se notifica a la interfaz que actualmente tiene el control activo y

esta debe ceder el control de forma activa. El registro de lectura correspondiente (40008) se activa tan pronto como se concede el control.

0x000	0x001	0x002	0x003
		16	

7.5.4.21 Liberar el control activo

Libere el control activo para que otra interfaz pueda enviar comandos al robot. Se puede acceder a los registros de lectura de datos y a los registros de lectura sin control activo. El registro de lectura correspondiente (40008) cambia a 0.

0x000	0x001	0x002	0x003
		32	

7.5.4.22 Cambiar al modo manual

Cambie al modo manual. En este modo, el robot solo se mueve mientras el usuario mantenga pulsado el pulsador de confirmación. Tenga en cuenta que para cambiar de modo es necesario realizar un reinicio de seguridad.

0x000	0x001	0x002	0x003
		64	

7.5.4.23 Cambiar al modo automático

Cambie al modo automático. Tenga en cuenta que para el cambio de modo es necesario un reinicio de seguridad.

0x000	0x001	0x002	0x003
		128	

7.5.4.24 Restablecimiento de seguridad

Restablezca el sistema de seguridad tras una interrupción. Para determinadas funciones de seguridad, el robot requiere una acción clara e intencionada por parte del usuario para continuar. Para comprobar si se requiere dicha acción, utilice el registro de lectura correspondiente (40009).

0x000	0x001	0x002	0x003
		256	

7.5.4.25 Iniciar la interrupción del movimiento

Tras una interrupción por motivos de seguridad, el robot puede seguir avanzando y continuar con su movimiento. Con esta propiedad, se puede volver a poner en marcha el robot. De lo contrario, se puede enviar un comando de parada para detener la aplicación en curso. Para comprobar si es necesaria dicha acción, utilice el registro de lectura correspondiente (40009).

0x000	0x001	0x002	0x003
		512	

7.5.5 Registros de lectura

7.5.5.1 Registro de estado

Utilice el registro de estado para leer el estado actual del robot. El estado se proporciona como un número de bit.

7.5.5.1.1 1. Registro de estado

Dirección absoluta: 40000 Dirección relativa: 0x000	
0	Fuera de servicio
1	Conectar
2	Cargando
3	Listo
4	En proceso
5	Calibración
6	En ejecución
7	Modo autónomo
8	En pausa
9	Se reanuda
10	Parada de seguridad
11	Parada de seguridad
12	Limitación de posición
13	Control manual (HGC)
14	Colisión
15	Parada de emergencia

7.5.5.1.2 2. Registro de estado

Dirección absoluta: 40001 Dirección relativa: 0x001	
0	Parada normal
1	Error
2	Desconexión
3	Parado
4.. 15	<i>Reservado</i>

7.5.5.1.3 3. Registro de estado

Dirección absoluta: 40002 Dirección relativa: 0x002	
0 ... 15	<i>reservado</i>

7.5.5.1.4 4. Registro de estado

Dirección absoluta: 40003 Dirección relativa: 0x003	
0 ... 15	<i>reservado</i>

7.5.5.1.5 Registro de errores graves

Dirección absoluta: 40004 Dirección relativa: 0x004	
0 ... 15	Información sobre la categoría de error.

El registro de errores graves lee los dos primeros bytes del identificador del error actual. Confirme el error a través del registro de propiedades para leer el siguiente error.
Encontrará más detalles sobre el significado de los valores del registro de errores en la asignación de los registros de errores.

7.5.5.1.6 Registro de errores menores

Dirección absoluta: 40005 Dirección relativa: 0x005	
0 ... 15	Información sobre el error dentro de una categoría.

El registro de errores menores lee los dos últimos bytes del identificador del error actual. Confirme el error a través del registro de propiedades para leer el siguiente error.

Encontrará detalles sobre el significado de los valores del registro de errores en la asignación de los registros de errores.

7.5.5.1.7 Registro de aplicación

Dirección absoluta: 40006 Dirección relativa: 0x006	
0 ... 15	ID de aplicación actual

El registro de aplicación lee el ID de la aplicación que se está ejecutando actualmente.

7.5.5.1.8 Registro de calibración

Dirección absoluta: 40007 Dirección relativa: 0x007	
0	Si el robot está calibrado

El registro de calibración indica si el robot está calibrado. 0 significa que el robot no está calibrado. 1 significa que el robot está calibrado. Tenga en cuenta que el robot debe estar conectado a su motor para proporcionar una respuesta significativa sobre la calibración.

7.5.5.1.9 Registro de control activo

Dirección absoluta: 40008 Dirección relativa: 0x008	
0	Si el canal Modbus tiene el control activo

El registro de control activo indica si el canal Modbus tiene el control activo. 0 significa que Modbus no tiene el control activo. 1 significa que Modbus tiene el control activo. El canal Modbus puede solicitar el control activo mediante 0x30002 0x04 Control activo.

7.5.5.1.10 Registro de interrupciones

El registro de interrupciones muestra las opciones de diálogo de interrupciones activas.

Dirección absoluta: 40009 Dirección relativa: 0x009	
0	Se requiere restablecimiento de seguridad
1	Es necesario continuar el movimiento

2	Es necesario cambiar al modo manual
3.. 15	<i>Reservado</i>

7.5.5.1.11 Registro de modos de funcionamiento

El registro de modos de funcionamiento indica si el robot se encuentra en modo manual (1) o en modo automático (0).

Dirección absoluta: 40010 Dirección relativa: 0x00A	
0	Si el robot se encuentra en modo manual

7.5.5.2 Registros de lectura definidos por el usuario

Dirección absoluta: 40016 Dirección relativa: 0x010		...	Dirección absoluta: 40255 Dirección relativa: 0x0FF	
0 ... 15	Datos definidos por el usuario		0 ... 15	Datos definidos por el usuario

Utilice los registros de lectura de usuario para recibir información de la aplicación. La función **modbus.read_user_register** de la aplicación lee los datos.

7.5.5.3 Registros de posición (articulaciones 1 a 6)

Dirección absoluta: 40256 Dirección relativa: 0x100		...	Dirección absoluta: 40261 Dirección relativa: 0x105	
0 ... 15	Ángulo de la articulación 1		0 ... 15	Ángulo de la articulación 6

Lea el valor angular de una articulación determinada en décimas de grado. Si se lee 101, el ángulo del eje es de 10,1°. El valor solo se actualiza si se escribe el indicador de lectura de posición en el registro de propiedades. Si se leen valores superiores a 32768 (=215), estos corresponden a valores negativos. Reste 32768 para obtener el valor absoluto. Un valor de 32869 corresponde, por ejemplo, a un ángulo de articulación de -10,1°.

7.5.5.4 Registro de coordenadas (x, y, z, roll, pitch, yaw)

Dirección absoluta: 40262 Dirección relativa: 0x106		...	Dirección absoluta: 40267 Dirección relativa: 0x10B	
0 ... 15	Coordenada x del TCP		0 ... 15	Coordenada de guiñada del TCP

Lea las coordenadas o la orientación del centro de la herramienta en el orden x, y, z, roll, pitch, yaw. Las coordenadas espaciales se leen en décimas de milímetro. Si se lee una coordenada de 314 para «
», el valor es de 31,4 milímetros. Las coordenadas de rotación se leen en décimas de grado. Si se lee

una coordenada de 271, el valor es 27,1 (mm o °). El valor solo se actualiza cuando se escribe el indicador Read-Position en el registro de propiedades. Si se leen valores superiores a 32768 (=215), estos corresponden a valores negativos. Reste 32768 para obtener el valor absoluto. Un valor de 33039 corresponde, por ejemplo, a un valor de coordenada de -27,1 (mm o °).

7.5.6 Asignación de los registros de error

La información de error siempre aparece en dos registros. El registro de errores menores contiene el número de error propiamente dicho. El registro de errores mayores contiene uno de los siguientes valores, que se asignan a diferentes categorías de error de la siguiente manera:

101	ADDONS
102	CONFIG
103	BASE DE DATOS
104	CONTROLADOR
105	EDITOR
106	BRIDGE
107	CONEXIÓN
108	DIALOG
109	DISTRIBUIDOR
110	EVENTO
111	CINEMÁTICA
112	SOCKET
113	NÚCLEO
114	SAFETY
115	AJUSTES
116	SIGNAL
117	ESTADO
118	SCRIPT
119	THREAD

201	BLOCK
202	COGNEX
203	CONTROL
204	ROS
205	MODBUS
206	REST
207	TCP

301	AX
302	CFG
303	CMK
304	CMD
305	CTL
306	ECAT
307	LIC
308	MATH
309	MCM
310	VARIOS
311	SISTEMA OPERATIVO
312	PPM
313	PYAPI
314	SIG
315	SYS
316	TLC
317	UTIL

7.5.6.1 Ejemplo

Si se leen los registros de errores [113, 4], esto se corresponde con el código de error CORE_004 generado por el robot.

7.5.7 Ejemplos de código Python

Estos ejemplos están pensados para su uso en un sistema externo que se conecta de forma remota con el robot.

7.5.7.1 Ejemplo 1

```
from pymodbus.client.sync import ModbusTcpClient

def main():
    # connect to client
    client = ModbusTcpClient("192.168.80.203", port=5020)
    client.connect()

    # read all status
    address = 0x000
    count = 4
    status = client.read_input_registers(address, count)
    print("STATUS", status.registers[0], status.registers[1],
          status.registers[2], status.registers[3])

if __name__ == "__main__":
    main()
```

Conectarse al servidor. Leer el estado del robot.

7.5.7.2 Ejemplo 2

```
from pymodbus.client.sync import ModbusTcpClient

def main():
    # connect to client
    client = ModbusTcpClient("192.168.80.203", port=5020)
    client.connect()

    # command to connect to motors
    address = 0x000
    values = [1 << 0]
    client.write_registers(address, values, unit=1)

if __name__ == "__main__":
    main()
```

Conectarse al servidor. Conectar el robot a sus motores.

Nota: El comando << se utiliza para el desplazamiento de bits.

Bit	Comando	Entero
0	1 << 0	1
5	1 << 5	32

7.5.7.3 Ejemplo 3

```
from pymodbus.client.sync import ModbusTcpClient

def main():
    # connect to client
    client = ModbusTcpClient("192.168.80.203", port=5020)
    client.connect()

    # command to run script with id 5
    address = 0x000
    values = [1 << 9, 5]
    client.write_registers(address, values, unit=1)

if __name__ == "__main__":
    main()
```

Conectarse al servidor. Ejecutar la aplicación con el ID 5.

7.5.7.4 Ejemplo 4

```
from pymodbus.client.sync import ModbusTcpClient

def main():
    # connect to client
    client = ModbusTcpClient("192.168.80.203", port=5020)
    client.connect()

    # read all error
    address = 0x004
    count = 2
    error = client.read_input_registers(address, count)
    print("ERROR", error.registers[0], error.registers[1])

    # acknowledge error
    address = 0x002
    values = [1 << 2]
    client.write_registers(address, values, unit=1)

if __name__ == "__main__":
    main()
```

Conectarse al servidor. Leer el mensaje de error actual. Confirmar el mensaje de error.

8. Módulo de cámara Cognex

8.1 Descripción general

El módulo de cámara Cognex le permite conectar una cámara Cognex preconfigurada a AX Portal. Una vez conectada, puede activar los trabajos preconfigurados en la cámara para capturar y procesar una imagen. A continuación, los datos procesados se transfieren a AX Portal, donde podrá leerlos en su aplicación.

8.2 Configuración de una cámara

Para configurar una cámara Cognex para que sea compatible con AX Portal, siga estos pasos:

1. Descargue e instale el software [In-Sight Explorer](#) de Cognex.
2. Conecte la cámara a la alimentación eléctrica y conéctela a su red. También puede conectarla directamente a su ordenador, siempre que su configuración de red esté establecida en una dirección IP estática.
3. Inicie In-Sight Explorer, conceda el acceso si se le solicita e instale los componentes necesarios (por ejemplo, Microsoft Visual C++, Microsoft .NET Framework).
4. A continuación, se abrirá automáticamente la vista Easybuilder del Explorer.
5. Vaya a «Sistema» → «Añadir sensor/dispositivo a la red». Busque su cámara y añádala a la red.
6. Asegúrese de que la configuración de red de la cámara sea compatible con la configuración de red del robot de la serie AX. Asegúrese también de que el puerto Telnet esté configurado en 3000.
7. Seleccione su cámara y haga clic en el botón «Connect» para conectarla al In-Sight Explorer.
8. Ahora ya está listo para crear tareas (por ejemplo, para calibrar la cámara y para detectar objetos que el robot debe agarrar) y guardarlas en la cámara. Encontrará más detalles en la documentación de Cognex.

8.3 Uso de una cámara

Para utilizar la cámara de Cognex en una aplicación de AX Portal, se requiere lo siguiente:

- El robot y la cámara deben estar en la misma red,
- el puerto Telnet de la cámara debe estar configurado en 3000,
- y la cámara debe estar en modo «online».

A continuación se muestra un ejemplo de cómo se puede utilizar un trabajo preconfigurado para detectar las coordenadas del círculo.

```
# modify these settings to fit to your application
camera_ip = "10.0.0.210"
job_name = "CircleDetection"
event_id = 1
keyword = "coordinates"

# connect the camera to the application
cognex.connect_to_camera(camera_ip)
# select the job to trigger
cognex.change_job(camera_ip, job_name)
```

```
# trigger the event and wait for a response
response = cognex.get_camera_message(camera_ip, event_id, keyword)
print(response)
# close the connection
cognex.disconnect_camera(camera_ip)
```