

Controlando el Robot Colaborativo ABB YuMi: De la Simulación al Control Externo con Python

Fase 1 - Programación y Simulación Virtual



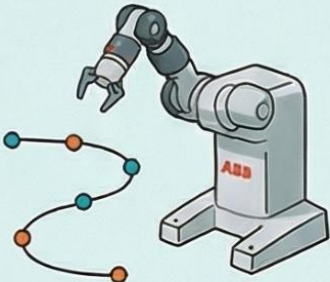
1. Instalar Software Esencial:
Se requiere RobotStudio para simulación y programación, y RobotWare como sistema operativo del controlador



2. Configurar la estación de trabajo:
En RobotStudio se crea un nuevo proyecto cargando el modelo IRB 14000 y se añaden los grippers a cada brazo.



3. Crear trayectorias visuales:
Moviendo manualmente los brazos del robot virtual, se guardan las posiciones clave para crear trayectorias auxiliares, por ejemplo, Home, Pos1 y Pos2.



4. Sincronizar con RAPID:
La trayectoria visual creada se convierte automáticamente en código RAPID, el lenguaje de programación nativo del robot, mediante la función “Sincronización con RAPID”.

```
MODULE Main
  PROC Path1
    PROC 1
      FROE 2
      PROC Ser
      END Patin
  END Path1
  PROC Path1
    ADE Steretre(1)
    ADL Path1
    END Patin
  END Path1
  ADE Path1
  PROC Rácaler(5, Y,Z)
  END Patin
```

Fase 2 - Conexión y Transferencia al Robot Real



5. Conectar Físicamente PC y robot:
Se establece una conexión directa entre la computadora y el controlador IRC 5 del robot utilizando un cable Ethernet.



6. Vincular controladores en RobotStudio:
Desde el software, se establece la conexión con el controlador del robot real, creando una relación para transferir datos del entorno virtual al físico.



7. Modo Automático :
El FlexPendant debe estar en modo operativo automático, para que RobotStudio pueda obtener acceso de escritura sin confirmación de usuario.



8. Solicitar acceso de escritura:
Se solicita permiso desde RobotStudio para poder modificar los programas del robot, lo cual debe ser concedido desde el FlexPendant.

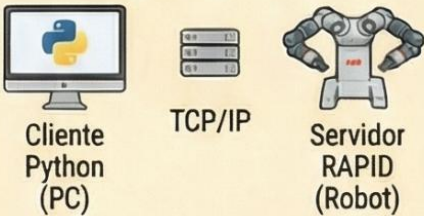


9. Transferir código al robot:
El código RAPID generado se envía desde el controlador virtual de la PC al controlador real del robot, dejándolo listo para ejecutar un rutina.

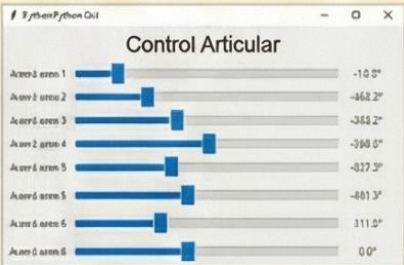
Fase 3 - Control Externo con Interfaz Python



Arquitectura de Comunicación Cliente-Servidor

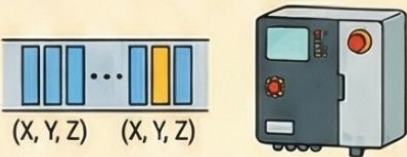


Opción A: Interfaz Gráfica para Control Articular

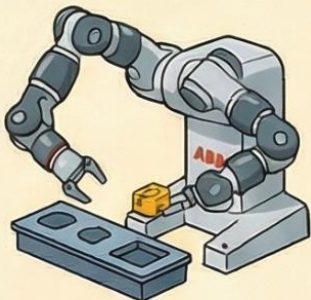


Control Articular (Tkinter)

Opción B: Optimización con Buffer de Instrucciones



Para evitar pausas entre movimientos, se implementó un sistema de cola (buffer) que permite enviar múltiples coordenadas cartesianas (X, Y, Z) de una sola vez.



Flujo de ejecución remota
El robot ejecuta secuencialmente las posiciones almacenadas en el buffer, logrando un movimiento continuo y eficiente, ideal para manipulación de objetos.