|  |  |
| --- | --- |
|  | AMBIENTE INTEGRADO DE APRENDIZAJE  DEPARTAMENTO DE INGENIERÍA INDUSTRIAL |

**MANUAL DE USO UR3**

**Caracterización general**

Para el funcionamiento del brazo por medio de Python usando la librería Socket es necesario comprender el funcionamiento básico del robot. Este funcionamiento se puede consiste en los movimientos y la configuración de las herramientas.

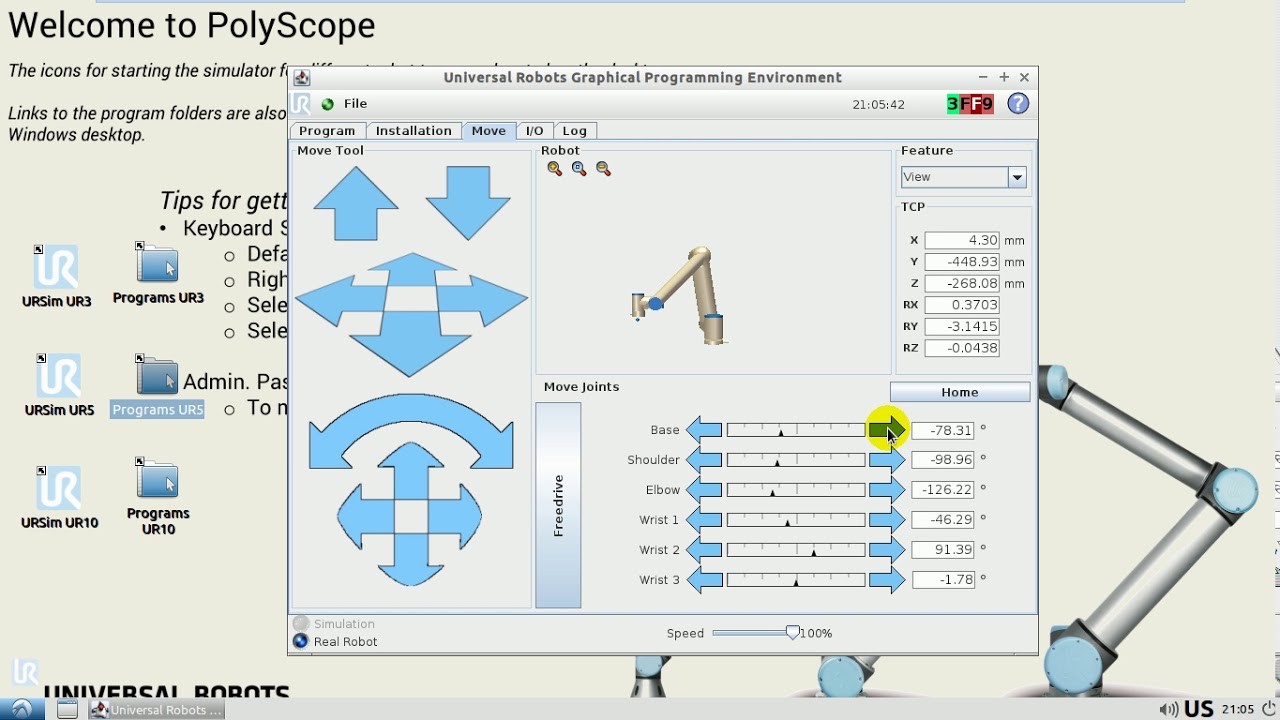
* **Movimientos**

El brazo tiene tres tipos de movimiento:

* MoveJ: Movimiento general sin ruta trazada
* MoveP: Movimiento con ruta trazada (no lineales)
* MoveL: Movimiento lineal del TCP (Tool Center Point)
* **Herramientas**

Las herramientas se configurarán generalmente de forma individual desde Polyscope; sin embargo, en el uso general se podrá usar la salida *Tool\_out[0]* para el control ON/OFF desde el puerto del brazo.

El movimiento en general se realiza por la configuración de los ángulos de cada unión del robot, definiendo posiciones específicas a las que se quiere llegar. Esto se observa en la imagen que se encuentra a continuación:



**Python - Socket**

Por medio del protocolo Socket (desde Python) se pueden enviar instrucciones de control directamente al brazo. Estas instrucciones deberán seguir un formato especifico. A continuación, se encuentra el proceso a ejecutar para el uso de esta funcionalidad:

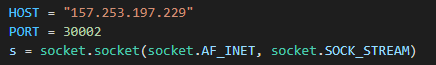
1. Configurar el servidor

Se configura el servidor como se muestra en la imagen inferior.



1. Configuración del usuario

Se configura el Host con la IP del robot (Se encuentra en la esquina izquierda superior en “*About”*).



1. Inicialización de la conexión

Se conecta el usuario por medio del puerto definido previamente.



1. Comandos

Para enviar comandos se utiliza la función *send( ).* Es importante codificar los mensajes a enviar en *utf8.* Un comando está dado por una función y el conjunto de parámetros que dicha función requiera. Ejemplo, la función ***movej*** requiere como parámetros una lista con las posiciones de los ángulos de las juntas, la aceleración y la velocidad del movimiento. Estas instrucciones deben enviarse junto con un tiempo de espera antes de continuar con la siguiente instrucción, ya que en caso de no esperar, se enviarán instrucciones sin que se completen las previas.



Para configurar alguna de las salidas digitales o de herramienta es necesario usar la función *set\_diigtal\_out( ).* Esta función requiere la salida a configurar y el estado. Cabe notar que para la salida de la herramienta es necesario configurar el voltaje de salida en 24V.



1. Documentación

* Comandos generales: <https://www.zacobria.com/universal-robots-knowledge-base-tech-support-forum-hints-tips-cb2-cb3/index.php/ur-script-send-commands-from-host-pc-to-robot-via-socket-connection/>
* Para las funciones revisar el archivo: “scriptManual.pdf”

**URScript**

Para usar actuadores de OnRobot como el gripper VG10, es necesario correr un archivo de URScript que ejecute a través de Python el código de la interfaz y los parámetros de uso. Para conseguir el archivo de UrScript se usará un *Magic File* de Universal Robots. Este archivo se utiliza de la siguiente forma:

1. Copiarlo en una USB en la carpeta base.
2. Insertar la USB en el *Teach Pendant* del Robot
3. Aceptar desde el *Teach Pendant* la advertencia respecto a la copia de archivos
4. Esperar hasta el final de la transferencia automática de archivos
5. Retirar la USB

Este proceso dejará en la USB los archivos de ejecución del URScript que haya creados para el robot. Se deberá abrir estos archivos en formato TXT.

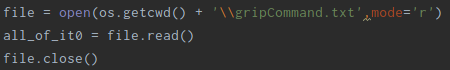
Para este caso se usará dos archivos, uno para el proceso de succión y otro para el de liberación. Cada uno de estos será un archivo de texto muy extenso; sin embargo, los parámetros a cambiar son pocos.

Inicialmente se debe configurar de qué forma se desea realizar la succión. Esto se configura en el archivo txt, en el que se define la función *vg\_grip\_delayed.* Esta función se configura al final del archivo y los parámetros que se deben modificar son:

* *Channel:* Este parámetro define qué canal se activa (0: Canal A, 1: Canal B, 2: Ambos)
* *Vacuum:* Este parámetro define el porcentaje de vacío (kPa) que se ejecuta por el robot. El máximo valor posible es 80 kPa. Se recomienda usar entre 20 y 40 para cajas de entre 0.2 y 1 kg.



Una vez se configuran estos parámetros, se debe importar el archivo y leerlo línea a línea, guardando el texto.



Finalmente, se debe enviar la señal al robot con toda la información leída. Esto ejecuta la acción de succión.



Se adjunta un ejemplo en el archivo socketGrip.py

Es muy importante recordar para el uso del gripper la distribución de los canales, ya que se debe modificar la disposición de las gomas según se desee usar. Si se cierran los brazos, la succión de cada canal quedará en forma pseudo-diagonal, por lo que su uso será limitado.

