

Trabajo durante 2023 UR3e

Félix García Fernández

Introducción

En este documento se muestra un breve resumen del trabajo realizado en los dos meses en el CIC LAB. El objetivo de este documento es que una persona que no haya trabajado en el UR3e pueda entender y continuar con este a partir de la lectura del documento. Todo el código nombrado está comentado en Github en el siguiente repositorio https://github.com/Felix-Garci/UR3e_com.

Funcionamiento de la comunicación

Para comunicarse con el UR3e desde Python se envían unos registros desde el programa hasta el robot. Por lo que en ambos lados debe haber un programa y se deben conectar. Para ello se utiliza un script en Python que se detalla a continuación. Para la comunicación se tienen que formalizar unos registros previamente. Después formalizar la comunicación y finalmente que el programa que está en el lado del robot interprete estos registros de forma adecuada.

Registros

Para estos registros se debe mirar el archivo "control_loop_configuration.xml" en el que nos vamos a encontrar tres secciones. La primera key="state" está destinada a variables que pasan del robot al ordenador. La segunda key="setp" está destinada a las variables que pasan del ordenador al robot. Y la tercera key="watchdog" está destinada a la sincronización de ambos. Dentro de estas secciones se pueden añadir o eliminar variables y se las puede asignar un tipo en concreto.

Comunicación desde Python

Para la comunicación desde Python se está usando una librería que nos proporciona Universal Robots para usar con sus robots. El código tiene varias funciones que se agrupan en 2 grupos. El primer grupo sería el init en el que se inicializan las variables importantes y las propias del cálculo de errores y procesos matemáticos. Por otro lado, está la función connect que se encarga de iniciar la comunicación con el robot y todas las demás son para la gestión de las peticiones que podremos hacerle al robot.

En la función connect es donde estos registros que se han preparado anteriormente se le mandan al robot para que entienda que es lo que tiene que devolver y que es lo que va a recibir. Además, se prepara el watchdog.

Una vez conectado entra en el thread de la conexión. Este thread estará corriendo la función monitor. Esta función se trata de un bucle en el que está diciéndole al robot que espere hasta que le pida algo. Tiene tres estados. Por defecto entra en el estado 0 que es el de reposo hasta que se le dice que entre en otro estado. EL 1 es para pedirle que valla a unos valores en ángulo de sus articulaciones y el estado 2 es para darle una posición.

Para poder cambiar de estado hay unas funciones `go_ang`, `go_pos` que reciben como parámetro un ángulo o una posición y cambian de estado para que el robot ejecute esta orden. Además de estas funciones hay una para pedir los ángulos en ese momento y otra para pedir la posición de ese momento.

Finalmente hay una función de stop para poder cortar la conexión y que se salga del bucle de monitor. Además de cerrar el thread.

Comunicación desde UR3

Para la comunicación desde este lado se requiere un programa en el lenguaje específico del robot. Este programa tiene dos partes. Before start y Robot program. En el primero se instancias las variables y las configuraciones iniciales. Mientras que en el segundo es el bucle de funcionamiento principal. En este bucle, se crean tres posibles estados. Espera, mover a referencia de ángulos, mover a referencia de posición cartesiana. Cuando el programa de Python se comunica con el programa del polyscope se le manda un vector que contiene la referencia ya sea angular o de posición y un valor más para determinar el estado.