

Resumen extendido

En este Trabajo Fin de Grado, se ha incorporado un *Framework de Desarrollo de Robótica* (RFS) llamado ROS (*Robotic Operating System*) en un sistema avanzado robótico de asistencia a la movilidad (SARA), diseñado en el Departamento de Electrónica durante los últimos años.

La Figura 1 muestra el diagrama general de SARA, que será necesario incluir en el modelo de comunicación de ROS, de modo que sea posible usar después las múltiples funciones de navegación autónoma sobre SARA.

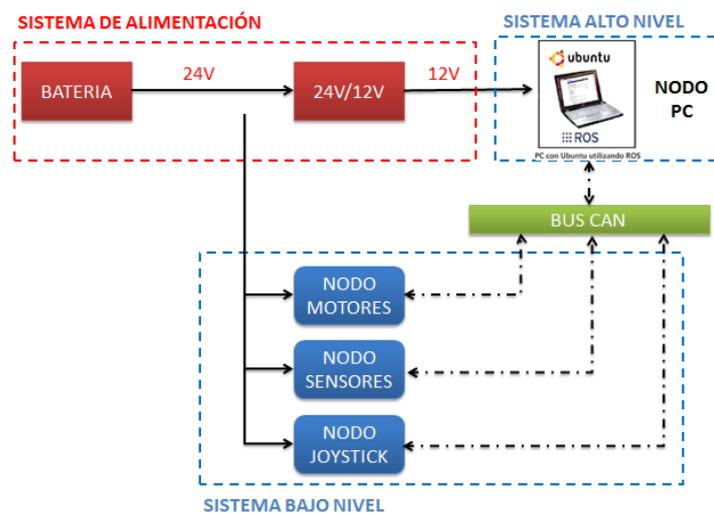


Figura 1 - Esquema general de SARA

ROS

Actualmente ROS es utilizado por una amplia comunidad de diseñadores de aplicaciones robóticas, ya que permite trabajar en diferentes lenguajes de programación (C++, Python o Lisp), está basado en una estructura distribuida y es un software de código abierto.

La finalidad de este proyecto, es incorporar ROS en la silla de ruedas SARA, de cara a que en un futuro se consiga implementar un sistema de navegación autónomo con el uso de las funciones de ROS.

Para poder entender los distintos módulos implementados en este Trabajo Fin de Grado, es necesario hablar sobre ROS. A parte de sus principales características, es necesario conocer los conceptos fundamentales para poder trabajar con esta plataforma, así como las herramientas que incluye.

Una vez conocido como funciona ROS, se modelará la silla de ruedas en esta plataforma, con el fin de poder visualizar a SARA tanto en simulaciones como en la navegación en tiempo real. Este modelo se creará mediante el modelo URDF, modelo basado en el lenguaje XML.

Con el modelo creado, se realizará una simulación de SARA en uno de los simuladores 2D más usados en ROS: STDR. En esta simulación, haciendo uso del teclado se podrá dirigir la silla por el mapa y visualizar los valores que leen los sensores de ultrasonidos, sin necesidad de disponer físicamente de la silla.

Hardware

SARA dispone de varios módulos (motores, joystick y sensores) que se comunican entre sí mediante el bus CAN. Los distintos nodos envían mensajes por este bus, cada uno con su propio identificador. Para poder comunicarse con la silla de ruedas, es necesario disponer de un adaptador USB-CAN e implementar un nodo en ROS que realice esta comunicación.

A nivel hardware, para el desarrollo de este TFG se usará todo el hardware de bajo nivel con lo que cuenta actualmente SARA, procedentes de anteriores proyectos realizados por el Departamento de Electrónica de la Universidad de Alcalá, centrándose este TFG en el rediseño de la estructura física con el fin de mejorar su estabilidad, la distribución de los sensores de ultrasonidos y la interfaz con el usuario.

El punto de partida hardware es, por tanto, una silla de ruedas eléctrica a la que le han sido añadidos distintos elementos y circuitería electrónica. Entre otros cabe destacar dos encoders incrementales acoplados a los ejes de los motores, una tarjeta de potencia para controlar cada uno de estos dos últimos, sensores de proximidad, acelerómetros, un ordenador y una pantalla táctil.

La silla de ruedas así construida dispone de varios nodos, comunicados entre sí mediante el bus CAN. Los nodos que actualmente tiene la silla son:

- **Nodo joystick:** Formado por un joystick analógico, un display, un potenciómetro, varios botones y un teclado.
- **Nodo motores:** Formado por los motores, encoders y una tarjeta de potencia.

- Nodo sensores: Formado por sensores de ultrasonidos colocados alrededor de la silla y un acelerómetro para detectar colisiones.

Resultados

El paquete correspondiente al modelo de SARA se llama "sara_urdf", el cual genera y permite visualizar su modelo URDF. (Figura 2).

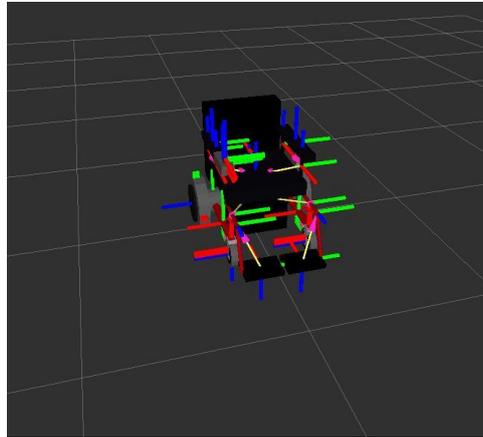


Figura 2 - Modelo URDF de la silla de ruedas

Para la simulación en ROS, se creará el paquete "sara_sim_stdr". Ejecutándolo se podrá dirigir la silla por el entorno de simulación obteniendo una lectura de los sensores de ultrasonido que hay distribuidos por la silla. (Figura 3).

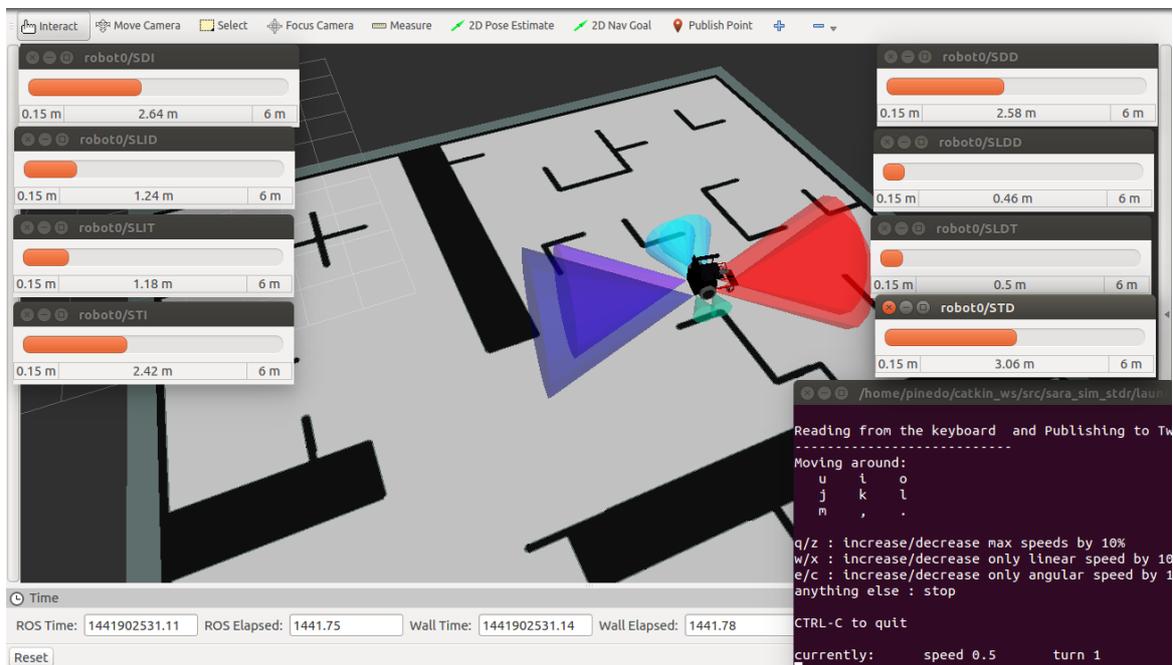


Figura 3 - Simulación de SARA en STDR

La comunicación entre el bus de comunicaciones CAN que usa la silla de ruedas y ROS se ha implementado en el paquete "canusb". Este se encarga de leer todos los datos que circulan por el bus CAN y de publicarlo en sus correspondientes *topics* (Figura 4)

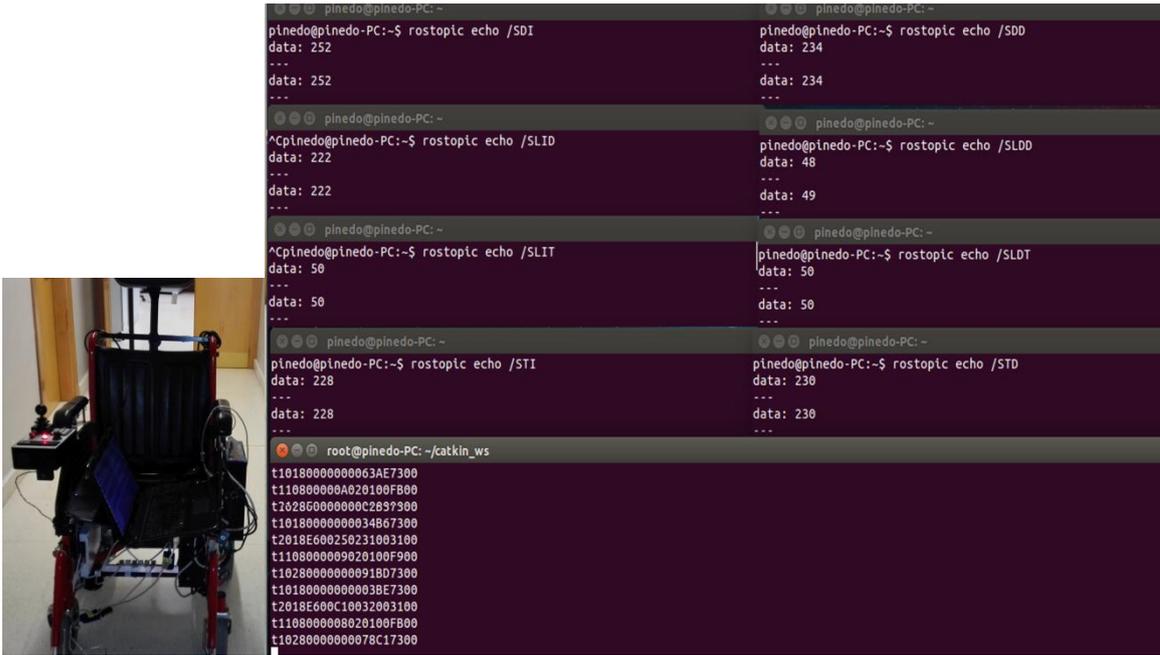


Figura 4 - Resultados comunicación CAN-ROS