Paquete de ROS 2

Repositorio caddy_ai2_ros2_control_hardware_curtis_motor_driver:

El repositorio caddy_ai2_ros2_control_hardware_curtis_motor_driver implementa la integración de un driver para motor Curtis dentro de la arquitectura ROS 2 control, permitiendo gestionar un caddy eléctrico (carrito de golf) desde un sistema robótico.

La comunicación con el motor se realiza a través de CAN bus (Controller Area Network), lo que asegura robustez y fiabilidad en la transmisión de comandos. El sistema está diseñado para trabajar con una frecuencia de control de 25 Hz, garantizando un comportamiento estable en tiempo real.

Entre sus características principales se encuentran:

Interfaz Hardware en ROS 2: el repositorio define un hardware interface plugin capaz de interactuar con el controlador Curtis, permitiendo la publicación y recepción de comandos de velocidad.

Soporte para CAN en Linux: incluye instrucciones para configurar interfaces USB-to-CAN, crear interfaces virtuales (vcan), y gestionar permisos via /etc/sudoers o grupos de usuario.

Dependencias clave:

libsocketcan y libsocketcanpp para manejar la comunicación CAN en C++.

can-utils, herramienta común en Linux para monitoreo y pruebas sobre buses CAN.

Soporte específico para el adaptador Ixxat USB-to-CAN V2 compact.

Facilidad de prueba: se pueden crear interfaces virtuales (vcan_motor_drv o vcan_steer_drv) para simular el bus y publicar comandos con ROS 2 sin necesidad de hardware físico.

Gestión en ROS 2: el driver se lanza como un controller dentro de ros2_control y permite enviar comandos, por ejemplo:

En conjunto, este stack permite controlar el motor eléctrico de 3.3 kW del caddy a través del controlador Curtis 1234, gestionando modos de operación y facilitando la integración con arquitecturas robóticas modernas.

Revision #2
Created 25 February 2026 23:02:13 by Rafael Carbonell Lázaro
Updated 25 February 2026 23:10:52 by Rafael Carbonell Lázaro