File: servo\_node.py

Author: Andrew Burroughs

Detailed Description:

This file implements the node controlling the servo motors for the excavation bucket. The servo motors are controlled by an Arduino Uno and required the output pin to be set on the Jetson Nano. The Arduino detects when the pin is set from low to high and executes the preprogrammed sequence of commands without user interaction.

Issues:

There is no way to stop the servos from moving while the program is running.

Software Documentation

Global Variables

**servoStateSubscriber**

This is a ROS2 subscriber of type Bool that looks for a ROS2 message of topic servo\_state and executes the servoStateCallback function when it receives a message.

**currentValue**

This is a Boolean value that tracks the current value of pin 18.

Function Documentation

**\_\_init\_\_**(self):

This function is called when the file is first run. It creates an instance of the super class with the parameter servo\_node, creates the servoStateSubscriber, then sets the GPIO pin 18 as output with initial value of LOW.

Expected Input:

None

Expected Results:

This function should create a ROS2 node with name servo\_node, creates the subscription, and sets the pin to LOW.

**servoStateCallback**(self, msg):

This function is called when the node receives a ROS2 message of topic servo\_state. If the msg.data is True, the node will set pin 18 to high and set the currentValue to True. If msg.data is False, pin 18 will be set to LOW and currentValue will be set to False.

Expected Input:

msg – ROS2 message of type Bool

Expected Results:

The node will set the pin to the value corresponding to the data from the msg and cause the servos to move when changing from the low to high values.

Change Log:

7/3/2022: Documentation was created