TESTING – US Localization (Orientation) Testing

Project: Design an Autonomous Robot

Task: To design an autonomous robot that is capable of navigating to a predetermined position while avoiding obstacles and firing objects at two targets. This is to be done in the shortest time possible.

Document Version Number: 1.0

Date: March 12th

Tester: Adrian Pan

Software Version: 1.0

Hardware Version: 2.0

Goal: To determine which method the robot should use to find its orientation

Table of Contents

PURPOSE 3

OBJECTIVES 3

PROCEDURE 3

EXPECTED RESULTS 3

FORMAT OF OUTPUT REQUIRED 3

CALCULATIONS 4

TEST REPORT 4

ACTION 4

DISTRIBUTION 5

GLOSSARY 5

# PURPOSE

The purpose of the test is to know which method is more accurate in determining the location of the robot. (Rising edge or falling edge). This information is important as it will help to better locate the robot in the competition. This test is also made to know the accuracy of the better method, so that it might be improved if there is a big range of error.

This test will be done with the hardware version 2.0 and version 1.0 of the code that can be found in the “Localization Test” folder.

# OBJECTIVES

The objective of the test is to know which localization methods will work better.

This test is a simulation of the localization part of the competition. Since the competition will be held at an indoor place, there will not be any factors that affect the measured distance of the US. Therefore, it is correct to assume that doing the test in the lab room can represent the real situation.

# PROCEDURE

1. Place the robot at arbitrary positions in the start area
2. Start the robot’s rising-edge method
3. After the robot determines its orientation, record the measured degree and actual degree using a protractor
4. Repeat step 1 and 3 ten times.
5. Repeat step 1 and 3 ten times with falling edge method

# EXPECTED RESULTS

The expected result is that both methods will be really close in terms of precision. Because tuning still needs to be done to determine wheel radius and width, precision is more important accuracy. The best case of the test is that the standard deviations of the both methods are small. The worst case is that standard deviations from both methods are really large.

# FORMAT OF OUTPUT REQUIRED

This table can be found in the “Localization Data.xlsx”.

# CALCULATIONS

The mean value can be obtained by the following formula:

EX: For the mean in rising edge,

For rising edge: For falling edge:

Mean: 4.1 Mean: -2.8

The standard deviation can be obtained as

For rising edge: For falling edge:

Standard deviation = 2.885 Standard deviation: 2.201

# TEST REPORT

The collected results are very positive, as the both rising and falling edge methods can be considered precise, as shown by the standard deviation values, 2.885 and 2.201 respectively. However, falling-edge method performs better than rising-edge method due to having smaller mean value and smaller standard deviation. One reason that falling-edge method is superior is that the speed at which the sensor polls distance. Rising- edge implementation faces the wall for the majority of the rotation while falling-edge faces away from the wall for the majority. This means that the ping sent during rising edge will return faster than that of falling edge. Every time the sensor sends out a ping, it will sleep for a short period of time before it collects distance data. In rising edge, the ping actually returns during the sleep period. This means that the robot will rotate a bit more before it realizes that it has reached the threshold. In falling edge, this problem is non-existent due to the longer distance as ping returns after the sleep period.

# CONCLUSION

The falling edge method is more reliable than the rising edge method in terms of standard deviation. Mean value shouldn’t be taken into consideration because wheel radius and wheel width are not optimized yet. A final conclusion will be made once the location test is conducted.

# ACTION

This test report should be keep within the software team in order to bring adjustments to the localization in the future. There will be a test on the location of the robot in the future.

# DISTRIBUTION

This testing belongs to the software development.

# GLOSSARY

US = ultrasonic sensor