

Testing Report

Yu Hong a1616861

3/11/2013

1 Introduction

The test cases covered in the report are relevant to the communication between PC side and Robot side. The files involved in this test are PCComms.java and RobotCommunication.java, which are located at

`https://version-control.adelaide.edu.au/svn/sep2013-2/individuals/Yu/HostSide/src/hostSideCommsUnit/PCComms.java`

and

`https://version-control.adelaide.edu.au/svn/sep2013-2/individuals/Yu/RobotSide/src/robotSideCommsUnit/RobotCommunication.java`

, respectively.

The tests are written in Junit and located in

`https://version-control.adelaide.edu.au/svn/sep2013-2/individuals/Yu/test/RobotTest.java`

2 Test Description and Rationale

The test cases regarding the communication is to test whether the value received by the robot is the one sent by the PC and to test the connection status between PC and the Robot.

1. Test the connection status after PC attempted to establish the connection.
2. Test the disconnection status after PC attempted to disconnect.
3. Test whether the robot can receive the movement value that is sent by the PC.
4. Test whether the robot can receive the sensor values that is sent by the PC.
5. Test whether the robot can send values that is over 100,000 to PC.

Test one will compare the connection status with the expected status after PC attempted to establish the connection. As with test one, test two will compare the connection status with the expected status after PC attempted to disconnect.

In terms of test three, as the movement commands that is sent from PC to Robot are integers ranged from 1 to 5, the test method will send integer within this range to the robot, and the robot will directly send the command it received back to PC. Therefore, the Junit test will compare the value that PC sends with the one it receives. As with test three, test four will test the sensor commands which are ranged from 11 to 19. The reason for the test five is that in some cases the robot needs to return large values over 100,000 and we need to ensure that the Robot is able to send back the value that is over 100,000.

However, the test for communication is complicated. In order to run each test case, the rest of the test cases should be blocked, otherwise the test would fail due to the interaction with other test methods. This JUnit test is able to pass all the test cases when it is running followed the above instruction.

A Test Code

The code below shows a part of the test codes from RobotTest.java, and only the @Test for Connection test cases are covered in this report for brevity.

```
1 @Test
2 public void testConnection() throws NXTCommException, IOException {
3     boolean connected = communication.isconnected();
4     Assert.assertEquals(true, connected);
5 }
6
7 @Test
8 public void testDisconnection() throws NXTCommException, IOException {
9     boolean connected = communication.isconnected();
10    connected = communication.disconnection();
11    Assert.assertEquals(true, connected);
12 }
```