Test report
for EV3 Robot Kit

Issued for: University of Adelaide

Authors:
Thomas McAtee (a1608768)
Xiang Guo (a1077337)

Decemeber 19th 2014

Contents

1 Introduction

This report was created to test the functionality of the new Lego EV3 Robot kit acquired by the University of Adelaide for potential use as part of future Software Engineering Project course. The EV3 will be tested against the previously used Lego NXT Robot kits and judged on their feasibility to replace the older NXT Robot.

The following sections will provide a brief overview on some of the differences between the NXT and the EV3. For more details please refer to th rest of this document. **TODO:** add images of both devices to the intro

1.1 Brick Comparison

1.2 New Features

TODO: Add images of new modules (e.g. wifi, USB host port, medium motor, improved ultrasonic sensor, etc.)

1.3 Improvements

2 Hardware

2.1 EV3 Brick

2.1.1 Display

- Resolution almost doubled
- Allows for better exception feedback

2.1.2 Ports

- Still using RJ-* connector; possibly proprietary? **TODO:** research
- Possibly not enough ports for the multitude of sensors available? (Possibly the best problem to have?)

2.1.3 Connections

2.1.4 Battery

- Lithium ion (AA battery pack also supplied)
- Much improved, especially given the nature of the new modifications (WIFI/Unix based OS)

2.1.5 Other

2.2 Sensors

2.2.1 Light

Same device as the NXT Robot. New API on EV3 which returns a float instead of an int. Accurate to 2 decimal places. New API does not allow calibration. The NXT API did not do a true calibration anyway, it simply created upper or lower bounds and normalised the sensor reading in software. If users wish to do this kind of soft calibration, they can use the source code for the NXT LeJOS API as a guide.

Has two modes, reflection and ambient. Reflection is the default mode and should be used for line following code. Black lines on normal print paper show up at around 40 percent reflection whereas white paper shows up as roughly 75
Ambient mode can be used to detect light sources. This might be an interesting feature if the robot can orientate itself based on a light source.

2.2.2 Color

Has three modes, Reflection, Ambient and Color. Ambient and Reflection modes are the same as the Light Sensor equivalent, however the sensitivity of the sensor is different.

Ambient mode is around half the sensitivity of the Light Sensor which means it should be able to differentiate between relatively bright light sources but might have trouble differentiating low level light sources.

Reflection mode is much more sensitive than the Light Sensor. Black lines on standard paper at 10mm range returned a reflection result of around 10 percent whereas white paper at the same distance returned 70 to 80 percent. This increased sensitivity should help in the project for Line Following and should be able to better handle smudges and imperfections on the test maps if the user implements a suitable tolerance in the code.

The color mode is meant to differentiate between up to 8 different colors, black, gray, red, green, yellow, blue, magenta, white. Testing was done using a laser printed standard 80gsm A3 paper, which contained a 256 color RGB chart. All tests were conducted at a range of approximately 5mm and at varying angles between 60 to 120 degrees. The sensor was able to pick up solid primary colors without issue at the angles tested. By primary we mean any colour where it's RGB hex value was represented by only pairs of F or 0. Eg. FFFFFF for white, FF0000 for red, FFFF00 for yellow etc.

Intermediate colors produced mixed results depending on the angle. Eg. FF8700 can be recognised as red at certain angles and as yellow at other angles. This is due to the reflection of the floodlight used by the sensor and the reflectivity of the ink on the paper. Therefore it is recommended in future projects to use only primary colors and non reflective paper to reduce the likelihood of erroneous readings. Other precautions that can be taken is to use colors that a true primary RGB eg Black 000000, white, FFFFFF, Red FF0000, Green 00FF00 or Blue 0000FF only and calibrate the sensor to only return one of those options.

2.2.3 Gyroscope

This new sensor is amazing when actually working. It will report either changes to the angle, or rate of change to the angle of the sensor with a decent amount of accuracy (plus or minus a degree or two).

During testing it was noted that an error that causes a sensor drift occasionally appears where the sensor reading will slowly increment or decrement. Using the Lego test suite, this issue could be solved by unplugging and replugging the sensor from the port. However, using the standard LeJOS test suite, unplugging the sensor resulted in an exception which it was not able to recover from even when sensor is replugged in. A restart of the brick may be required if this bug appears. This will be something of concern if future students wish to incorporate the sensor in a project. This appears to be a firmware issue, and not related to LeJOS, future firmware releases may address this issue. Please note that this bug will appear straight away when the sensor is activated, it will not occur mid operation. So if the sensor is properly started and checked when the brick is turned on, it should not present a problem later during other operations. It is then advised to have a gyroscope test in LeJOS before starting any main programs.

Possible expansions: -the different modes the sensor can be in -possible use of gyros to measure non-horizontal axes

2.2.4 Ultra Sonic

Much more accurate than the NXT sensor. Testing against different objects under 40cm distance against a tape measurer revealed a much improved sensor in the new kit. The EV3 US sensor had much higher fidelity and returned measurements in 1mm increments although the accuracy can have a margin of error between +- 4mm when detected surface was front on to the sensor. The older NXT US sensor was only able to report in 1cm increments and much higher margin of error of +- 3cm.

The minimum range of the sensor was also much improved. The NXT sensor loses accuracy within roughly 12cm. EV3 sensor was able to maintain accuracy to within 4cm.

The new sensor's cone of detection remains relatively narrow, no different to the NXT version. And both sensors suffer decreases in accuracy when the detected surface is at a sharp angle. The EV3 sensor was able to maintain accuracy when detected surface is at no more than 30 degree angle to the sensor(where 0 degree means completely front on to the sensor).

2.2.5 Bump

2.2.6 Backwards Compatibility

All NXT sensors can be used on the EV3 Brick with out issue. However, EV3 sensors will not be recognised by the NXT brick.

2.3 Motors

2.3.1 Large

2.3.2 Medium

The rotation sensor is a motor with less torque than a Large motor (typically used to move the robot) but with higher speed due to it's smaller diameter (it is capable of 260 RPM rather than the Larges 175).

Possible problems: depending on the application it may be very important to keep track of the 'arm' being moved by the motors position. In tests it proved very easy to cause the shovel arm (design included in the Lego build manual) to collide with the motor itself which could potentially damage the internal mechanics of the motor.

(insert image of collision here)

To counteract this, it may be necessary to 'reset' the position of the arm during device shutdown (this method description implies that the maintained 'position' of the motor will be reset upon a device reset)

The Rotation Sensor utilises a motor port, which as part of the project shouldn't be a problem as students will only be provided with three modules that can utilise the four ports.

2.3.3 Backwards Compatibility

TODO: has this been tested??

2.4 Building blocks

TODO: What exactly are the building blocks???

3 Software

3.1 Eclipse

Functionality of the Eclipse plugin (pushing code from a PC to the block) has been vastly improved
TODO: add image describing the simplicity of pushing code to the EV3 block

3.2 Java RMI

- Explain what the RMI model involves
- Explain what this enables an EV3 programmer to do
- Give an example of a code block for this

Show examples, demonstrate different paradigms (e.g. setting one device as the 'default' block vs keeping a hold of several blocks)
TODO: example of obtaining a device connection in new leJOS compared to old

3.3 leJOS

Previously the leJOS Java Virtual Machine has been utilised within the SEP course in order to provide an interface for programming the NXT using the Java programming language. However as well as solid functions the leJOS NXT library had incomplete, untested and unimplemented methods, all of which handicapped SEP project development.
Given the incomplete nature of the previous library we were at first apprehensive, but given exposure to the leJOS EV3 library we are very satisfied with the progress that has occurred since the previous library.
TODO: can I say we?

- New features supported
- QWERTY onscreen-keyboard is a little unwieldy

4 Connectivity

A core feature of the NXT was its connectable nature; it could be programmed and controlled by a wide range of heterogeneous hardware and operating systems (including Windows, OSX and other Linux-based systems). The EV3 has built upon this fundamental property by introducing a new connection method (wifi based connections) and by improving upon existing connections (for example by increasing the speed of USB connections enormously and by streamlining Bluetooth connectivity).

- Due to architecture change you can connect directly to the block via an IP address

4.1 WiFi

- Plug and play functionality (supports several mainstream chipsets)
- leJOS has a 'connection' menu item
- Connections can be configured on the Micro SD image to avoid repeated configuration

4.1.1 OSX

Tested, functional

4.1.2 Windows

4.2 Bluetooth

4.2.1 OSX

- Still unable to simply connect via bluetooth
- Forum posts show that other users' efforts are similarly problem affected

4.2.2 Windows

4.3 USB

- Device now has a USB host port (supports 3 levels of daisy chaining)
- Addition of standard USB port helps facilitate the use of a WIFI dongle and also allows for USB storage
- The new USB speed is a blistering 480 Mbit/s (compared to the NXTs 12 Mbit/s)

4.3.1 OSX

4.3.2 Windows

5 Conclusion

5.1 Feasibility

5.2 Recommendations

Line following design ideas

5.3 Additional project ideas

6 Revision History

Version	Date	Author	Description
1.0	15 Dec 14	Xiang Guo (a1077337)	Created Draft 1

TODO: Synchronise this with GitHub revision history