**SYSTEM DOCUMENT**

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: 2.0

Date: March 13th, 2015

Author: James Fuh, Yutian Zheng, Gwyneth Pang

**TABLE OF CONTENTS**

System Model…………………………………………………………………….…1

Hardware Available and Capabilities……………………………………………..2

Software Available and Capabilities………………………………………………3

Compatibility…………………………………………………………………………3

Reusability…………………………………………………………………………...3

Structures…………………………………………………………………………….4

Methodologies……………………………………………………………………….4

Tools…………………………………………………………………………………..4

Glossary of Terms………………………………………………………………......4

**SYSTEM MODEL**

Preliminary Flowchart of the Robot's Operation

Initialization

Navigate to the target

Arrived at the target

Yes? No?

Stop

Finish recording

**HARDWARE AVAILABLE AND CAPABILITIES**

NXT Brick:

- 32-bit ARM7 microprocessor

- Bluetooth wireless communication (Bluetooth Class II V2.0 compliant)

- USB 2.0 full speed port

- 4 input ports, 6-wire cable digital platform

- 3 output ports, 6-wire cable digital platform

- 100 x 64 pixel LCD graphical display

- Loudspeaker

- Powered by 6 AA batteries

Motor:

* Capable of turning/rotating the robot in a desired angle if needed

Ultrasonic sensor:

* Able to measure a distance between 0 to 255 cm with an error of +/- 3cm
* Works better on rigid surfaces



Touch Sensor

* An analog sensor that can detect when the sensor’s red button has been pressed and when it is released.
* Can be programmed to action using three conditions—pressed, released, or bumped



Color Sensor

* Digital sensor that can detect the color or intensity of light that enters the small window on the face of the sensor
*  Can be used in three different modes: Color Mode, Reflected Light Intensity Mode, and Ambient Light Intensity mode.

Electromechanical limitations

Require 6 1.5V AA batteries to run.

The motor runs at 160–170 rpm, with a running torque of 20 Ncm and a stall torque of 40 Ncm.

**SOFTWARE AVAILABLE AND CAPABILITIES**

The robot will be running in a Java virtual machine called leJOS NXJ. It is capable of running things in object oriented languages such as Java, which offers several functions like array, synchronization, and exceptions.

**COMPATIBILITY**

The mechanical component of the robot should be very sturdy, and there should be no compatibility issues with the mechanical design. The software design however, will have compatibility issues. The variables defining the same thing, should be the same, and the threads should not overlap. Furthermore, the display should not be accessed by two separate methods at the same time. We also need to be careful of racing conditions.

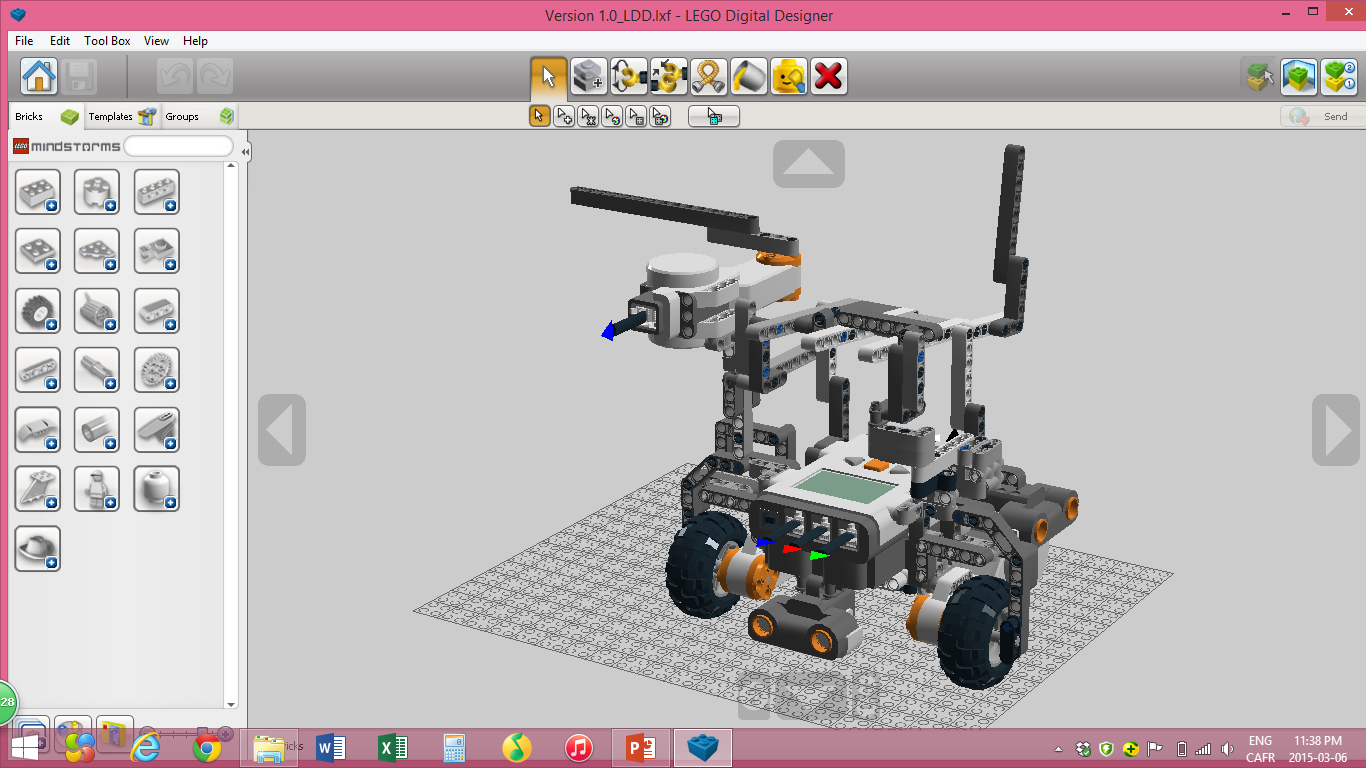
.

**REUSABILITY**

Over five labs during the design course, several parts of the core code can be reused on the robot to perform the designated task. The odometry code, the navigation code, and the localization code allow the robot to display the current coordinate and travel to the desired position. The ballistic code for lab 5 allows the robot to perform the task of launching the ball in order to hit the target. (Refer to the requirements document for more details)

**STRUCTURES**

The mechanical design includes the launcher, the loader, and the chassis. They are tied together using Lego pieces and elastic bands. Furthermore, we have installed two ultrasonic sensors and one light sensor on the chassis. Refer to the LDD design in the Dropbox for more details.



**METHODOLOGIES**

Our methodology will be to divide the whole project into several subtasks, and then assign them to the group members depending on their capabilities. There will be meeting sessions every week so that we can keep track of the progress. We will also evaluate the feasibility of the ideas during each meeting session to see if there is any room of improvement.

**TOOLS**

Physically we have the NXT toolkit which allows us to build a robot. The robot will be able to perform specific tasks with the assist of the software, which will be mentioned below.

Lego Digital Designer allows us to visualize how a specific model is built in 3D so that we can build the robot physically from there. Java is the core part of the software as it provides different functions and algorithms which allows the robot to function in a specific behavior.

**GLOSSARY OF TERMS**

N/A