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: 3.0

Date: March 22nd, 2015

Author: James Fuh, Yutian Zheng, Gwyneth Pang, Eric Vuong

Contents

[System Model 1](#_Toc415944056)

[Hardware Availabilities and Capabilities 2](#_Toc415944057)

[Software Availabilities and Capabilities 3](#_Toc415944058)

[Compatibility 3](#_Toc415944059)

[Reusability 4](#_Toc415944060)

[Structures 4](#_Toc415944061)

[Methodologies 4](#_Toc415944062)

[Tools 5](#_Toc415944063)

[Glossary of Terms 5](#_Toc415944064)

# System Model

Figure 1 shows the mechanical design of the robot as of week 4. It includes two ultrasonic sensors, one light sensor, and three motors connected to one NXT brick. A LDD of the design can be found in the folder.

The light sensor is used by the odometer correction and the light localization. Only one sensor is needed in this case, because they detect the same thing.

The ultrasonic sensor is used by the ultrasonic localization, and the obstacle avoidance part of the navigator. Only one ultrasonic sensor is needed for the localization. However, since the robot had empty ports after implementing the other parts of the system, the team decided to use two ultrasonic sensors for the obstacle avoidance in order for the obstacle avoidance to be more accurate, since it will encompass a wider range of situations.

The motors are used by the odometer, which is used by the navigator, the launcher, and the display. The first two motors are used to allow the robot to travel, and the launcher uses the third motor in order to shoot the Ping-Pong balls into the targets. In lab 5, one of the sub teams were able to make the robot load and shoot properly using only one motor rather than two, thus their design with slight modifications in order to make it more durable will be implemented.

# Hardware Availabilities and Capabilities

The NXT brick is limited by the following:

* 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

The motor is only able to rotate clockwise or anticlockwise. Since there are two motors, both motors need to work together in order to achieve the desired result. If they are not calibrated to turn the same speed, the path traveled will deviate from the desired path. For example, if the desired path is straight, the left and the right motors need to rotate forwards at the same speed. If the left motor rotates faster than the right motor, then the robot will turn towards the right. Furthermore, the motors are powered by an electromagnetic force, which generates the torque that rotates the motors.

Ultrasonic sensor:

The ultrasonic sensor is able to measure a distance between 0 and 255 cm with an error of approximately 3cm. It also works better on rigid surfaces since sound will bounce off the surface more uniformly in order to have a more accurate reading.

Color Sensor

The color sensor is able to detect the color or intensity of the light that enters the small window on the face of the sensor. It has a flood light which will be able to help filter the colors that it detects. A differential filter will be created, which will allow it to detect changes in color, rather than absolute colors. This will allow it to detect the black lines without depending on the environment of the room.

# Software Availabilities 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

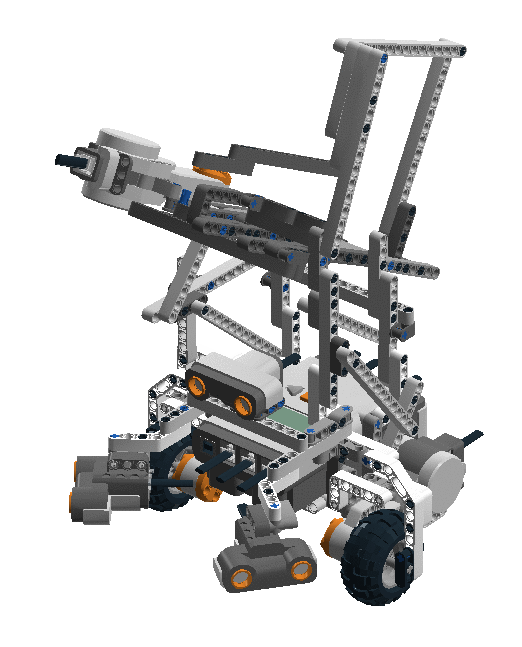
Refer to the compatibility section of the requirements document.

# Reusability

Refer to the compatibility re-use section of the requirements document.

# Structures

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



# 

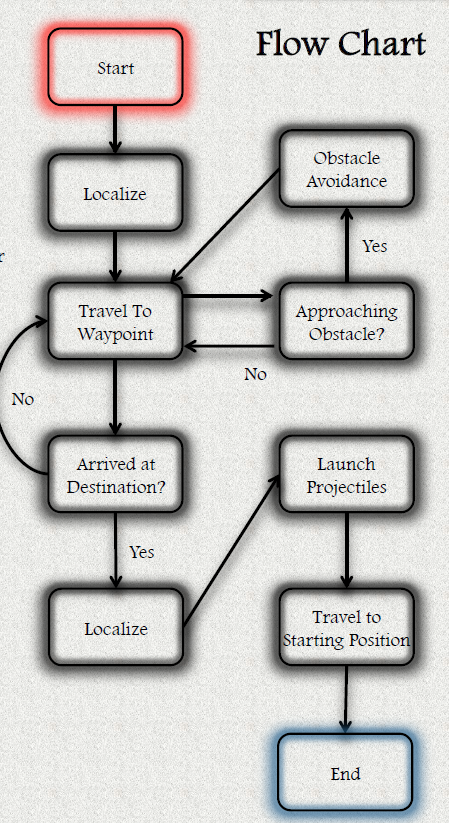
# Methodologies

Two fundamental strategies the team used to achieve the goal of the objectives. The first focus is to make a design that is as independent as possible of each other. Therefore, there will be no possible way of delaying another task if the previous one is not complete on time especially for the software department. For instance, the localization development process of the software architecture will not affect the progress of the launching development. Second, most of the team’s decision making will be based on test results to know whether or not to reject a proposal, software algorithm, or hardware. In this way, the team will not only make decisions based on intuition but also by experimental results to better reflect the constraints and uncertainty in the NXT hardwares.

Tasks are separated via the Gantt chart. Mainly, one member will be responsible for the documentation, two for mechanical design, and three for software design. However, in case of tardiness on the critical path, resources will be reallocated to make sure the team finishes according to plan. The tasks have been separated according to each member’s preference as much as possible, and the testing will be allocated to the person who designed the part in order to speed up the process. For example, the member responsible for the launching mechanical design will be responsible for constructing the launcher, the testing, and the documentation. Since they will have the most knowledge about the system they designed themselves, they will be able to analyze the data much faster, and able to correct the mistakes more accurately.

Every week, there will be a meeting with the professor, the TA, and the team members. **Refer to the capabilities document for more details**. During the meetings with the professor and the TA, the progression will be the main discussion. In comparison, the meeting with the team members will mainly be comprised of the division of tasks, updates, discussions of new ideas, and future plans and decision making.

Here is the flowchart of the robot’s steps that the team has decided to use in order to achieve the objective. By following these procedures, the testing (refer to Simulation and Integration test Under Tests of week 7) have confirmed that the robot does indeed perform the tasks in the shortest time possible.



# Tools

Physically, the NXT toolkit, which is used to build a robot, is available to the group. The robot will be able to perform specific tasks with the assist of the software, which will be mentioned below.

Lego Digital Designer (LDD) allows us to visualize how a specific model is built in 3D so that the robot’s build can be saved as a checkpoint. 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. Also, Java is a great object-oriented programming, thus allowing the user to easily understand and use user-made methods to further improve the communication.

# Glossary of Terms

N/A