Autonomous Object Sorting Robot

Submitted by:
Rick Fernie, James McLaughlin, Adam Nans, Alejandro Turla
Great White Robotics, San Diego, CA

Submitted to:
John Kennedy and Lal Tummala
Design Co. Ltd, San Diego, CA
# Table of Contents

1. Abstract

2. Problem Definition
   - Customer Requirements
   - Functionality of Autonomous Robot
   - Conceptual Design

3. System Design
   - Background Research
   - System Description
   - Functional Diagrams
   - Identify Constraints

4. Project Plan
   - Task Decomposition & Scheduling and Project Management
   - Budget
1. Abstract

In many situations, autonomous robots can provide effective solutions to menial or dangerous tasks. In this case, it is desirable to create an autonomous robot that can identify objects and relocate them if the object meets certain criteria. Obviously, when dealing with a large number of objects, this is a very menial task. This is an excellent application for a robot of this type. In this case, to keep costs and design complexity low, the robot is designed around the iRobot Create platform and uses several different sensors to collect information about the robots environment to allow the robot to react accordingly.

2. Problem Definition

Customer Requirements

The problem we are attempting to solve is to create an autonomous robot that can navigate within an area, identify objects based on customer specified criteria, and then sort by relocating them to a specific location. The customer can specify the sorting criteria prior to operation.

Functionality of Autonomous Robot

1. Navigate 5’ x 10’ table without getting stuck in pockets
2. Identify desired ball locations using ultrasonic signals
3. Obtain status information through IR signal
4. Capture balls, identify color, and release if not correct color within 10 seconds
5. Move ball in to closest pocket if identified as correct
6. Avoid other robot and walls, as well as avoid sinking black ball early
7. Sink all balls of our color (identified prior to match), then sink black ball

Conceptual Design

A main microcontroller will handle the behavior of the robot using input from peripheral microcontrollers that will handle location and object information. Another peripheral microcontroller will control the arm used to capture balls, as well as record information from color sensor mounted inside a cup at the end of the arm. All circuit boards and assemblies will be mounted to robot via an aluminum plate mounted on standoffs in pre-tapped locations on robot.

3. System Design

Background Research

We researched several different areas to provide ourselves with sufficient information to decide on the most effective ways to collect data on the robots surroundings. For the color sensing, we looked into using single photodiodes with different color LEDs, as well as small camera vision systems and RGB color sensors. Ultrasonic receivers can either use amplified signals going to the A/D of the MCU, or comparing them to a reference, creating a simple pulse train entering the MCU. We also researched various motor types to control the arm used to capture balls, as well as possible alternate ball capturing techniques.

System Description

Object identification for movement will be accomplished through a set of four Sharp IR sensors, mounted appropriately to differentiate between balls, walls, and the other robot effectively (Figure 1). The center sensor will be the main object detection sensor, while the left-
and right-most will detect other robots, as well as helping to zero in on ball locations. The top-
most sensor will mainly detect walls, but can also identify other robots. A dedicated
microcontroller will record distance information for each sensor and provide the main
microcontroller with any necessary information.

![Figure 1: IR Sensor Layout (not to scale)](image)

Pocket location will be continuously recorded using four ultrasonic receivers. A set of
amplifiers will condition the signals from the receivers to be useful for a dedicated
microcontroller. The microcontroller will constantly calculate the robots location from the
pockets and identify the closest pocket. This information will be available to the main
microcontroller on request.

The arm used to capture and identify balls will also have its own dedicated
microcontroller. At the end of the arm will be a cup that will isolate the ball from any
surrounding light to increase the chances of correct color identification. An RGB color sensor
and LED(s) will shine on the ball to identify the color. The microcontroller will also handle
movement of the arm based on color identification, and will be initially triggered by the main
controller to capture a ball once the robot is oriented properly. It will provide a signal that tells
the main microcontroller whether it should move towards a pocket (meaning a correct color ball
has been identified), or will it automatically release and tell the main microcontroller to search for another ball.

**Figure 2: Conceptual Drawing**

**Functional Diagrams**

**Figure 3: Input/Output Diagram**
Identify Constraints

Due to the nature of the competition, the only real constraints placed on the robot are the rules of the game and the size of the play table. The robot is limited to the 5’ x 10’ area of the table. The rules of the game are:

a. The robot has to wait until a “go” signal is sent to begin play
b. A penalty signal will stop the robot if a penalty is incurred
c. The black ball must be sunk last or the game is automatically lost
d. The robot must release any balls not of its designated color for the match within 10 seconds
e. The first robot to sink all of its own balls (or the robot with most balls sunk by the end of the match) wins
4. Project Plan

Task Decomposition & Scheduling and Project Management

See attached Gantt chart.

Budget

<table>
<thead>
<tr>
<th>Item #</th>
<th>Description</th>
<th>Qty.</th>
<th>Unit Cost</th>
<th>Total Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>iRobot Create Programmable Robot</td>
<td>1</td>
<td>$129.99</td>
<td>$129.99</td>
</tr>
<tr>
<td>2</td>
<td>iRobot Create Battery w/Charger</td>
<td>1</td>
<td>$69.99</td>
<td>$69.99</td>
</tr>
<tr>
<td>3</td>
<td>iRobot Create Self-Charging Home Base</td>
<td>1</td>
<td>$59.99</td>
<td>$59.99</td>
</tr>
<tr>
<td>4</td>
<td>Sharp IR DP2D12</td>
<td>4</td>
<td>$14.95</td>
<td>$59.80</td>
</tr>
<tr>
<td>5</td>
<td>Ultrasonic Receivers</td>
<td>4</td>
<td>$1.00</td>
<td>$4.00</td>
</tr>
<tr>
<td>6</td>
<td>Peripheral MCU’s (PIC)</td>
<td>5</td>
<td>$1.50</td>
<td>$7.50</td>
</tr>
<tr>
<td>7</td>
<td>Main MCU (AVR)</td>
<td>1</td>
<td>$8.50</td>
<td>$8.50</td>
</tr>
<tr>
<td>8</td>
<td>Arm Motor</td>
<td>1</td>
<td>$10.00</td>
<td>$10.00</td>
</tr>
<tr>
<td>9</td>
<td>RGB Color Sensors</td>
<td>3</td>
<td>$5.80</td>
<td>$17.40</td>
</tr>
<tr>
<td>10</td>
<td>Arm Building Materials</td>
<td>1</td>
<td>$10.00</td>
<td>$10.00</td>
</tr>
<tr>
<td>11</td>
<td>PICKit 2</td>
<td>1</td>
<td>$34.99</td>
<td>$34.99</td>
</tr>
<tr>
<td>12</td>
<td>AD623 Instrumentation Amplifier</td>
<td>4</td>
<td>$1.71</td>
<td>$6.84</td>
</tr>
<tr>
<td>13</td>
<td>Engineering Costs (4 Engineers, 46 days)</td>
<td>4</td>
<td>$29,440</td>
<td>$117,760</td>
</tr>
</tbody>
</table>

Total Projected Cost $118,179.00