

Robotic Controlled Character Recognition

Jerrod Sutton

(Mentor: Dr. Jeff Gray)

Using a Lego's Mindstorms™ RCX Robot

Mountain Brook High School

Java Programming



Software Composition and Modeling Laboratory

SoftCom



Department of Computer and Information Sciences
University of Alabama at Birmingham

Background: RCX and leJOS

Abstract

Embedded systems represent over 90% of all microprocessors and exist in many devices used in daily activities (e.g., cell phones, avionics, automobiles, televisions). Such systems are typified by limited resources (e.g., small memory footprint), which make programming their behavior a challenge. Robotics control systems provide an excellent platform for investigating aspects of embedded systems design.

This project describes an investigation into a robotic control system that is a of recognizing written characters. The implementation of the project is based on a Lego Mindstorms robot that is controlled by software written in Java. The software for the robot was written in Eclipse, which is a popular Java development environment. The compiled bytecode is executed on a tiny virtual machine, which also must be downloaded onto the robot through an infrared connection.

The investigation extended the Ledeen-Teitelman algorithm, which partitions a character into a 3x3 matrix. For implementation on the robot, the algorithm was adapted to make a robot traverse the matrix while looking for differences read by a light sensor (i.e., a change in the sensor value indicates that a piece of a character exists in a specific cell of the matrix). During the traversal, whenever a dark value was read, a bit representing the cell location is stored. At the end of the traversal, the collected values of the light sensor represent a bit stream that can be mapped to a unique alphabet character. After traversal, the robot will display the character value on its LCD screen.

Project Objective

- To write a program in Java that will allow an RCX robot to identify letters according to a 3x3 matrix, and have the robot print the found characters on its LCD.
- The letters will be cut from black paper and mounted on a white backdrop. Each letter will be subdivided into 9 parts forming a 3x3 matrix.
- The robot will use the navigational code to steer its way through the matrix using transverse motions.
- The RCX robot will determine if part of the letter is in a particular section of the matrix by turning bits on or off according to the light reflected back into its sensor.

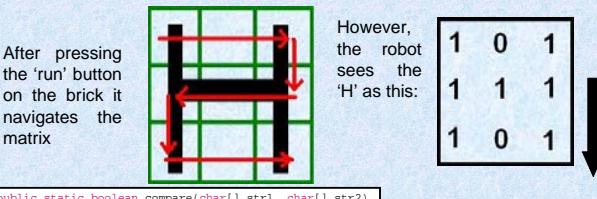


The wait times for the program to upload to the robot through the IR Port took forever.



The wait times for the program to upload to the robot through the IR Port took forever.

The Recognizer Algorithm

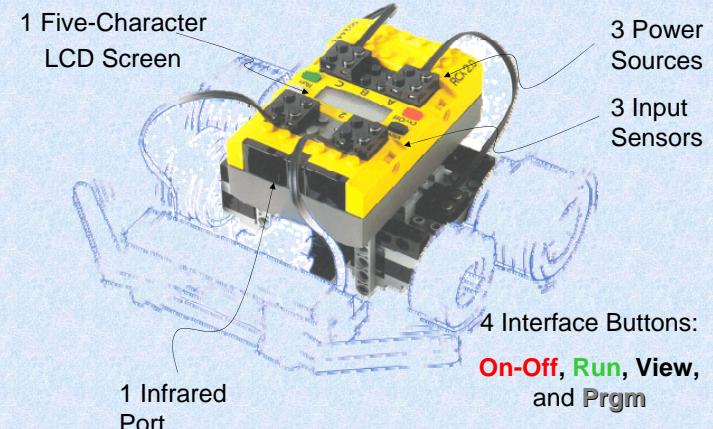


```
public static boolean compare(char[] str1, char[] str2)
{
    for (int i = 0; i <= 8; i++)
    {
        if (str1[i] != str2[i])
            return false;
    }
    return true;
}
```

Then its processed through this compare method



The alphabet array library where each of the 26 letters are depicted in their array form.



Obstacles of the Project

- The first hurdle of this project was learning Java and the Eclipse v3.0.1 environment. The project also required the use of leJOS, and I had to learn the libraries and interfaces provided by leJOS that interact with the RCX.
- The robot had a slight drift when it attempted a straight line.
- Manipulating the cut-out letters so that each letter will have a unique identity when read.
- Working with very limited memory in the brick, limited Java functions, and the excessively long downloading times.



Prints to the LCD Screen