

## SnackBot Sensor Report

Chris Shepherd

EEL 5666

7/7/05

## Table of Contents

Behaviors and Sensors.....	3
Bump Switches.....	3
IR Range Finder.....	4
Line Follower.....	4
Color Detector.....	5

## Behaviors and Sensors

SnackBot has four main behaviors. These are:

1. Avoid Obstacles
2. Follow a Line
3. Detect Color
4. Pick up and Object

Once SnackBot has received a command for a snack or drink he will follow a line to the kitchen. Along the way he will use his obstacle avoidance so as to not trip anybody who may be up and walking around the house. Once in the kitchen SnackBot will make his way to the pantry or refrigerator and select your item of choice by its packaging color. It will pick up your object and return it to you at the couch.

To accomplish these behaviors SnackBot will need bump switches, an IR range finder, photo reflectors, and a special color detector.

### Bump Switches

The Bump switches are attached to a bumper that wraps half-way around SnackBot. This allows the use of only four switches to detect bumps 180 degrees around the robot. One side of each switch is connected to a general I/O port pin which is pulled up internally. The other side is connected to ground. Thus when the switch is not closed the port pin reads a logic 1. And when the switch is closed the pin reads a logic 0. The program written uses these values to determine if it has run into an object, and if it has, on which side of the robot the object is. This is then used to change direction away from the object.

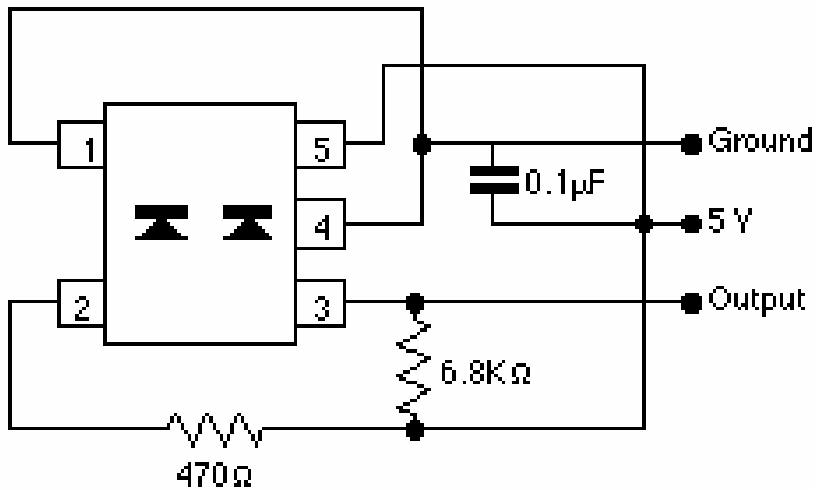
## IR Range Finder

The IR range finder is a Sharp GP2D12. It detects the distance to the nearest object and returns that distance as an analog voltage. This output is connected to an A2D capable port on the microprocessor. Using software based on some sample code from [bdmicro.com](http://bdmicro.com) I am able to convert this analog voltage to a digital number. To help avoid errors due to inaccurate data I take the average of 5 readings and use that as my sensor value. I do not know the threshold value that my program uses to determine if an object is close in front of it because this value changes every time the robot is run. This is because I wrote a calibration routine to determine this value each time the robot is powered up. This calibration helps reduce errors due to different lighting conditions.

## Line Follower

The line follower is made up of 4 Hamamatsu photo reflectors. I have not made the board that will hold all four photo reflectors, but I have tested the photo reflectors individually. Although the photo reflectors return an analog value like the IR range finder, no A2D conversion is necessary. This is because I am only interested in 2 states when following the line. I simply need to know whether the robot is on the line, or whether it's off the line. The line will be a high contrast line: a black line on a white surface. Simply needing to know whether it's on the line or not means only needing to be able to differentiate between black and white. The photo reflectors return a value of approximately 5 volts if placed over a black surface, and a value of approximately 0 volts if over a white surface. Since these are the logic levels of the microprocessor there is no

need for a conversion. The photo reflectors will be set up in the circuit seen below. The output will be passed to a general I/O port pin and read in through software.



### Color Detector

The color sensor consists of a red, blue, and green LED, and a single CDS cell. The theory behind this sensor is that different objects reflect light of different wavelengths. This is what causes an object to appear the color that it is. For example, a red object appears red in white light because it reflects the wavelengths of light that appear red, and absorbs all the other wavelengths. So, if the same object were placed in green light, it would still absorb all non-red wavelengths, but since this is green light and there are no red wavelengths to reflect, the object would not be able to reflect any light. I use this concept and measure the amount of light reflected under different lighting conditions to determine a color.

This sensor is made up of a CDS cell and three LEDs. A CDS cell varies in resistance based upon the amount of light it detects. The color sensor will use the LEDs

to shine only red, blue, or green light at a time. A red object will reflect all red light, causing the CDS cell to have a very different resistance than if a blue or green object, which will absorb red light, were present. This difference in resistance will be used in a voltage divider circuit to create different voltages to signal SnackBot if the object being tested is red, blue, or green.

The table below shows the different resistance values of the CDS cell with several objects under differing lighting conditions. These are most likely not the final threshold values I will use in my software as the resistance values are likely to change once the sensor is attached to the robot.

Object	Red LED	Blue LED	Green LED
None	130 k	240 k	300 k
Red	5.3 k	160 k	130 k
Blue	11 k	55 k	190 k
Green	10 k	130 k	80 k

With these values an easy algorithm to detect color has been determined.

Blue object if: resistance is less than 100 kOhms under blue light

Green object if: resistance is less than 100 kOhms under green light

Red object if: resistance is greater than 100 kOhms under blue and green light and  
resistance is less than 10 kOhms under red light