

Wassim Tawil
TAs: William Dubel
Steven Pickles
Instructors: A. A Arroyo
E. M. Schwartz

University of Florida
Department of Electrical and Computer Engineering
EEL 5666
Intelligent Machines Design Laboratory

JAMESBOT

Sensor Report

Table of contents

1. Introduction-----	3
2. Components-----	4 – 7
A. Bump switches-----	4
B. IR-----	5
C. CDS Cells-----	6
D. Voice recognition-----	7
3. Experimental Results-----	8 – 13
A. Bump switches-----	8
B. IR-----	8 - 11
C. CDS Cells-----	11 - 12
D. Voice recognition-----	12 – 13
4. Conclusion-----	14

Introduction

A robot without sensors is like a human without eyes; we need them in order to function properly. That's why it's crucial for a robot to have sensors if we want him to perform complex tasks. Different types of sensors can be used, depending on what we want the machine to do. For example, a robot that will look for a fire while avoiding obstacles, will probably need some IRs and a pyro sensor. In the case of JamesBot, four different sensors will be used: bump switches, IR, photoresistors, and voice recognition.

JamesBot is a voice controlled robot that will go into a room and search for a dark area to hide in. To accomplish this task, he will need different kinds of sensors. First, bump switches and IRs will be used for obstacle detection and avoidance. The photoresistors will help JamesBot find a dark place, and the voice circuit will be used to control him using audio commands. Below, I will go in depth on these sensors and how they were interfaced to the controller. I will also give the experimental results obtained.

Components

Bump switches

This is the least expensive and most basic sensor I'll be using. I actually got them for free for IMDL lab, and they will only be used as backups for the IR sensors. They'll be used to inform the microcontroller whenever the robot hits an obstacle. The bump switch is a digital sensor because it sends either a low or high voltage depending on the situation. It's basically a pull-up resistor, and whenever the bump is hit, it causes the switch to close and make the input low. At all other times, the input stays high. On my robot, I will use a total of three switches, on the front and on each side. I included a figure below to give you an idea.

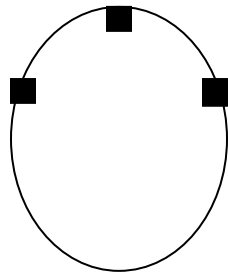


Figure 1

IR

I will use the Sharp GP2D12 sensors as they seem to be the most popular and provide good distance measurement. On JamesBot, they will be used for obstacle avoidance and wall following. The sensor has an emitter which sends out an IR pulse and a receiver which records the reflected pulse. Depending on the received pulse, the sensor can approximate the distance of the obstacle. The distance is reported as an analog voltage with a range of 4'' to 30''. I'm thinking of using three of these sensors; one in the front and two on the sides. An image of the sensor is provided below.



Figure 2: Sharp GP2D12

CDS cells

My robot will need to go into a room and hide in the darkest area. To be able to accomplish this, he will be equipped with photoresistors. Photoresistors are a type of light sensor, and can be described as variable resistors. Whenever there is a change in the light level, the resistance is changed. The weaker the light reflected, the greater the resistance, and vice versa. When it's connected to the microcontroller, the resistance will have to be converted to a voltage. This can be obtained by making a voltage divider circuit. An example can be seen below (acroname.com). If R1 is the photoresistor, then the voltage will increase with increasing light intensity.

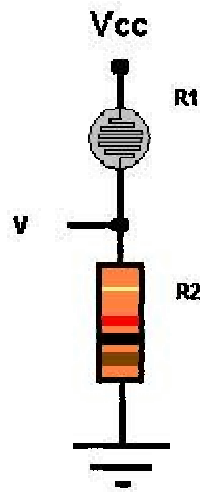


Figure 3: Voltage divider for Photoresistor

Voice Recognition

The most useful feature of JamesBot is its voice recognition system. It will be used to guide the robot to a specified location, and also in case his position needs to be changed while he's spying. I will be using the SR-07 Speech Recognition Kit from imageco.com. This kit has numerous features; it can handle twenty different words and can be programmed for either isolated or continuous speaking. It has non-volatile back up memory and can be easily interfaced to external circuits. When a trained word is recognized, the circuit outputs a digital number corresponding to this specific word. Since my robot will be controlled in places where I can't see it, I need to still be able to transmit my voice commands. This will be arranged by using a pair of 2-way radios, and placing one of them on the robot. Below I provided the SR-07's circuit.

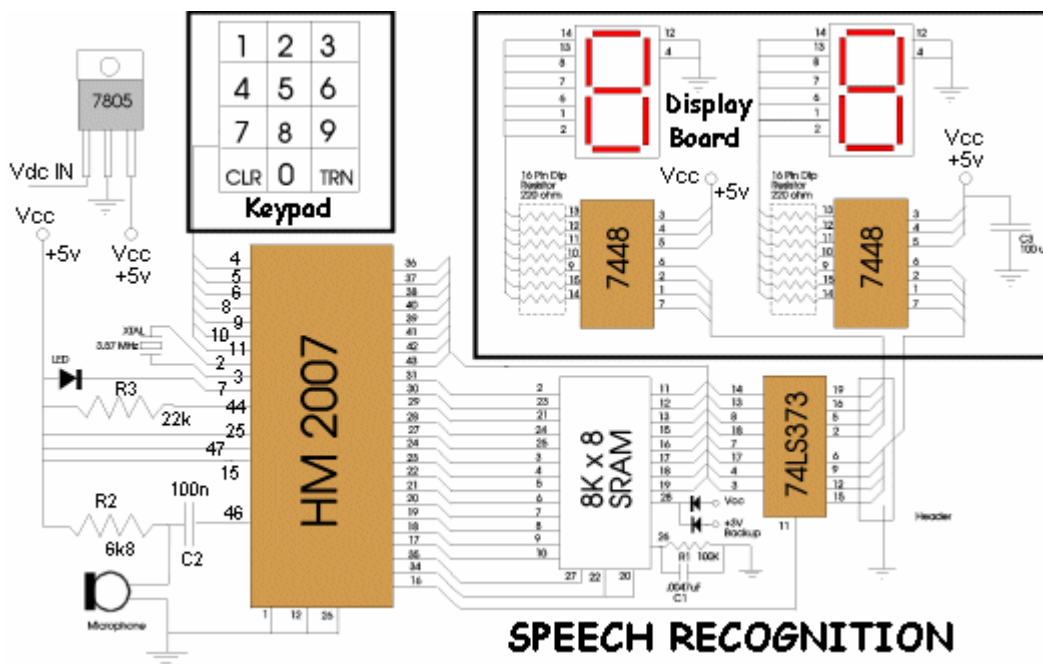


Figure 4: Speech Recognition circuit

Experimental Results

Bump Switches

I connected all my bumps to a common pin. I used a voltage divider circuit so that each switch will output a different value. Since this value was between 0 and 5V, I had to convert it to a digital value using the on board A/D converter. The Mavric is equipped with eight A/D pins located on Port F. I used Pin 0 for the bump switches. This is a 10 bit converter, so the outputted value was between 0 and 1024. Below I provided a table showing the values obtained when I tested my switches.

BUMP	DIGITAL VALUE
Right	678
Front	484
Left	302
Right + Front	750
Left + Front	580

Figure 5

IR

I used three sensors on my robot, one in the front and one on each side. I tested the IRs to see how well they work at different distances. I performed this test in a well lit

room, and I got values starting at one inch from the wall up to thirty five inches. I found out that the range at which these sensors work best is between four and twenty eight inches. Below that range, the values seem to increase very fast, and above that, they stop changing. JamesBot will not need to see that far; he will only avoid obstacles and do wall following which require up to six inches of detection, and in which case, these IRs work fine. For my obstacle avoidance, I programmed him to avoid anytime he gets a value greater than 300, and it worked perfectly. Below, I provided a table with all the value that I obtained in my tests, as well as a graph.

Distance (inches)	Digital Value
1	192
2	382
3	551
4	480
5	405
6	334
7	301
8	250
9	233
10	209
11	199
12	182
13	172

14	160
15	141
16	138
17	131
18	123
19	114
20	112
21	107
22	103
23	96
24	95
25	92
26	94
27	95
28	101
29	120
30	127
31	130
32	130
33	130
34	130
35	130

Figure 6

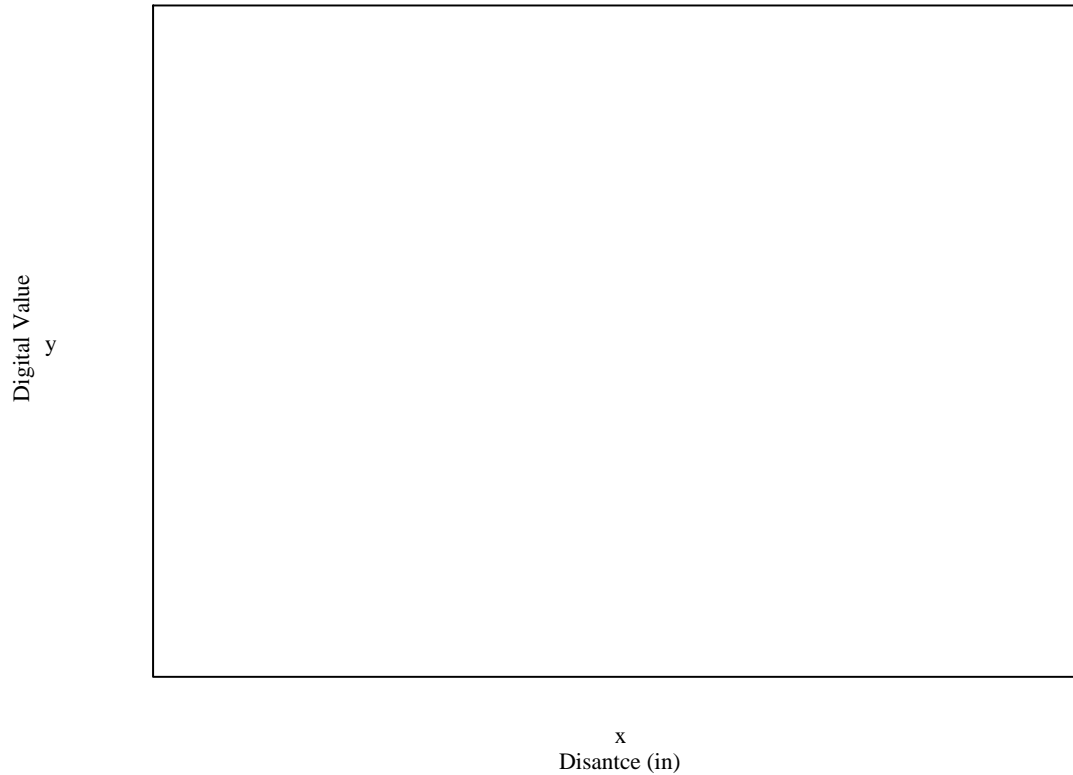


Figure 7

CDS Cells

I covered the cells with duct tape to isolate the light received. JamesBot is equipped with three photoresistors. One is placed in the front, and two behind the tires. I wanted to make sure his whole platform is in the dark before he stops; that's why I placed these sensors this way. Whenever all three sensors detect the dark area, he will stop wall following, and begin spying. If only one or two of the sensors detect the dark area, then he keeps moving, until all three agree. I tested these sensors in my room, and found all the typical places that JamesBot could hide under. I then placed him in these areas and

read the value. I got values under a chair, a desk, and under the bed. I connected the photoresistor to a voltage divider circuit with a 15Kohms resistor, and connected the changing voltage to a pin on the A/D converter. Here are the value that I obtained.

JamesBot position	Digital Value
Not hiding	640
Under chair	550
Under desk	450
Under bed	140

Figure 8

Voice Recognition

This was the most difficult sensor I worked on because it had many factors. While testing this sensor, I had to account for such things as the tone of voice, the word length, outside noise, and the placement of the microphone with respect to my mouth. Since I was using a two way radio to send the commands, my voice came out less clear, which meant I had to be even more careful with my choice of words. The circuit can recognize forty 0.96s words or twenty 1.92s words. I configured it to twenty so that I can get more choices for words, and because I will only be using about six or seven words. I first tested out the kit in my room with the door shut and without the two way radio. After trying many different words, I found out that the circuit works much better with words that have more than one syllable. For example, words like “left” or “right” were sometimes confused. I then used my two way radio instead, and noticed that the

recognition accuracy slightly decreased. It was still good enough though. I also found out that the distance between the microphone and the radio is very important. When I moved the radio away, the circuit would not recognize the commands as well. Another crucial factor is the tone of the voice. The accuracy got much better when each word was trained with a different tone of voice. The tricky part though is that every word had to be spoken with the same level of excitement as when it was trained. For example, if I trained the word “forward” in a happy voice and then repeated that same word with a sad voice, the circuit would not recognize it. So, to get the highest accuracy possible, the words need to have more than one syllable, they need to have a different tone, and they have to be repeated with that same tone, and the radio has to be kept the same distance from the microphone. I had some of my friends try to speak into the circuit to test if it recognized their voice, and it came out that it depended on the person. My brother, for example, who sounds like me, was able to get the circuit to recognize his voice most of the time. As for the other persons, the circuit would sometimes recognize their voice, depending on the word. The final test was to try the circuit while there was outside noise, like the television. The kit was placed inside the robot, so it was shut from outside interferences. When I spoke into radio though, and the noise was loud enough, the circuit would sometimes capture that noise mixed with my voice and get very confused. That was the worst case scenario, and even then, it still had more than fifty percent accuracy. So, to sum it all up, I was pleased with the performance of my speech recognition kit.

Conclusion

Overall, I was pleased with my sensor performance. The bump switches were the easiest to interface and they work with no error. The IRs work very nicely, as long as I keep them within the range of 4 to 28 inches. The photoresistors were enclosed in duct tape, so that the light coming from the ground was isolated. The only thing that will affect their performance is if I keep changing locations. If the room where I want them to work is dimmer than the one where I tested them, they won't work as well. The speech recognition was the one that gave me the most trouble. After training many words though, I figured out how to get the highest accuracy. The words spoken need to be distinct, have more than one syllable, and different tone. The microphone has to be kept the same distance from the radio at all times. The only disadvantage is outside noise. It doesn't perform as well while others are speaking.