Denise

An Autonomous Fire Fighting Robot

Proposal

By

Sean Justice

EEL 5666

August 4, 1999

Table of Contents

ABSTRACT

Denise is an autonomous robot that seeks out fire and signals where it is. Denise will be able to first ‘smell’ fire, and then proceed to find it. The robot used micro-switches, two different types of infrared detectors and a smoke detector to find the fire and maneuver through rooms. Through the integration of these sensors, Denise able to recognize when a fire was present and then safety maneuver and find the fire. When Denise found the fire, she then proceeded to signal where the fire was.

EXECUTIVE SUMMARY

The robot that I created was an autonomous fire-seeking robot named Denise. The robot is able to detect if there is objects in front of it; if it has bumped into a wall; whether there is smoke present and weather there is a flame in its range. To do this, I used a combination of sensors along with programming to be able recognize and react to each of the respected situations.

For obstacle avoidance, I need to be able to recognize if there was an object in front of the robot and how far it was. I also need to recognize whether the robot has run into an object, or if an object has run into it. To successfully accomplish this, I used two types of sensors, micro-switches and infrared emitters and detectors. Denise is equipped with six micro-switches. Three are located in the front and three are located in the back. They are able to detect if Denise has come into contact with anything. If the any of the micro-switches are closed, the robot will know that it has come into contact with an object and will react accordingly. Denise is also able to "see." Denise's "eyes" are actually four infrared emitters and detectors. They are located in the front and are positioned so that there are two on each side. The infrared sensors work by the emitters sending out modulated wavelengths, which travel through the air. The waves will reflect off of objects and be sent back to the detector. The detector returns a voltage that depends on the amount of intensity of the waves. By using the information provided by these types of sensors, the robot is able to autonomously wander about a room.

To be able to give the robot the ability to sense if there was smoke and to “see” a flame, I used two other types of sensors. To recognize that there was smoke present, the robot was equipped with a smoke detector. The smoke detector using ionization to recognize smoke particles. The smoke detector was integrated into the robot to alert it. The robot would not do anything until smoke was detected. After detection, the robot would wander around the room looking for fire. To find this fire, two infrared detectors were again used. These detectors were different than to ones than the ones used for obstacle avoidance. The infrared sensors detected wavelengths that were emitted from a flame. They were located in the front of the robot. With the used of these sensors and programming, the robot was able to recognize and react to smoke and fire.

INTRODUCTION

Background

Fighting fires is a very challenging and dangerous job. Because of its low pay and high risk, there are not nearly enough fire fighters to meet the demand. This is relevant in Florida, primarily in the summer when there are many brush fires the instantly brake out. Fire fighting is a very stressful job. Fire-people must be able to cope with high physical demands and high mental stress. They must be able to react quickly and efficiently to extinguish the fire in the shortest amount of time. They must make these quick choices when people’s lives are in danger.

Scope

The idea for my robot came from the stress that can come from fighting fire. My robot, Denise, could reside in buildings. When a fire breaks out, Denise can start to search and try to signal where the fire is located, while the fire fighters are on their way. After the fire fighters arrive, Denise could help them by searching other rooms for fire.

My robot will demonstrate the search the ability to recognize when a fire is present. It will then search and find the fire and signals were it is. To accomplish each of these behaviors I will use the following sensors; infrared emitters and detectors, micro-switches, and a smoke detector. A Motorola 68HC11 equipped sensor expansion board will control these sensors plus the control of the motors.

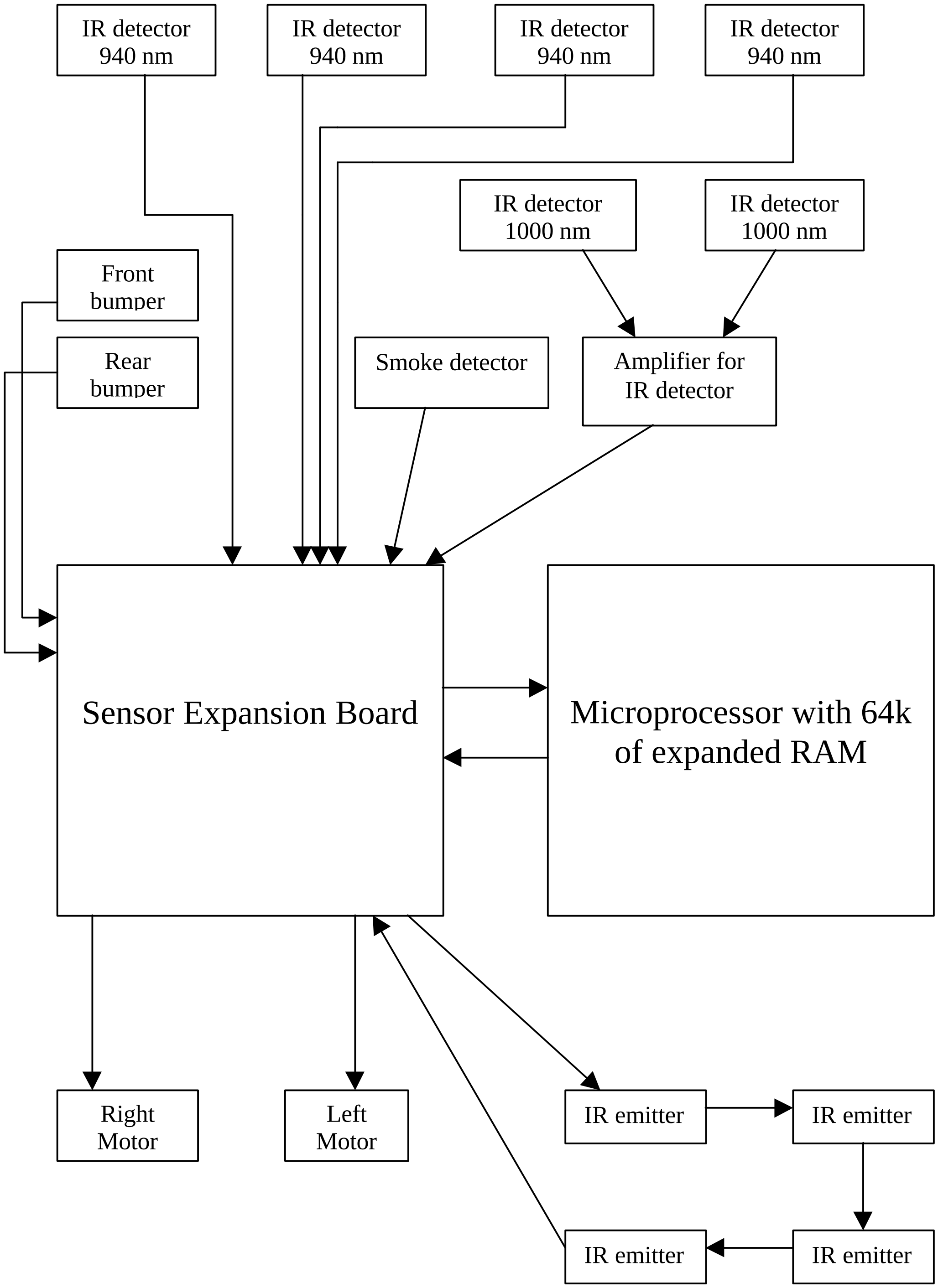
INTEGRATED SYSTEM

Description of System

The ‘brain’ of Denise is a Motorola 68HC11. The 68HC11 have been expanded so that there is 64k of programmable RAM. Megatronix makes the board that is used in this robot. The board is the same one used in Talrik, which is a robot made by Megatronix as well. The board is fairly small and provides more than enough RAM for my applications. Along with the microprocessor board, I also used a sensor expansion board. Megatronix also makes this board. The sensor expansion board had many features that were very helpful in the programming of the robot. Another board that was used in the robot was atwo-amplifier circuit used for the flame detectors.

Operation

The following block diagram shows how the integrated system works.



The robot uses eight analog ports, which the sensor expansion provides. The sensor expansion board actually has 20 analog ports, which makes it easy to add more sensors to the robot, if desired later. The microprocessor and sensor pack can be stacked to reduce room on the platform. This aspect was very nice, the compact size made to easier to place all of the sensors and battery packs on the platform.

MOBILE PLATFORM

Scope

The mobile platform of Denise had to be large enough to fit all of the sensors and devices that she would be carrying. I chose to design the platform myself and model it after the same platform used by the Talrik. There are some differences between them though. First, Denise's platform is larger, it is a 11-inch circular platform. Other differences include: three holes for wire passes, batteries are mounted on top of the platform rather that below, there isn't any bridge for a servo motor on top, there are only six mounts for infrared emitters. These modifications helped Denise's integrated system to work properly.

Specifications

When designing the mobile platform of Denise, I needed it to be able to have all of the integrated system fit properly and have the ability to be upgraded. The platform is designed to hold an extinguishing system, which was not installed during this semester. The extinguishing system is to be placed under the platform. This is why the battery packs are mounted on the top. Additions to the battery packs mounted on the top, the microprocessor plus the other sensors are mounted on the top. On the bottom of the platform the motors and the infrared emitters are mounted. The micro-switches are built into the platform. The platform also has a circular bumper attached to it. Figure one displays the top-view of the platform used for Denise. The platform turned out to meet all of the specifications to meet all of the objectives for Denise.

.

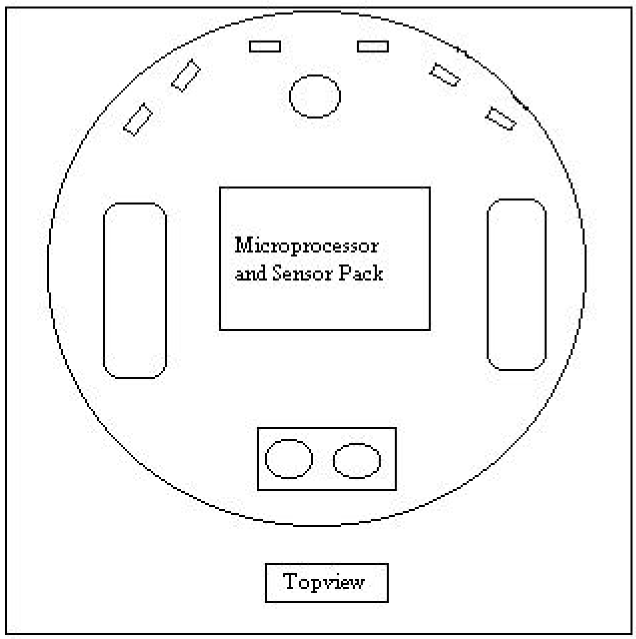


Figure One.

ACTUATION

Scope

Denise needs to be able to move about freely when looking for a fire. One of my specifications for the robot was that she should be able to rotate 360 degrees on its central axis. The motors used in Denise needed to have enough torque to move her throughout the room. For Denise's actuation two dc motors were used and one canister wheel.

Motors

The motors used for Denise's are actually servo-motors that have been hacked to perform as dc motors. The characteristics of the motors used in Denise are 42 oz./in. The hack was quite easy to perform. First the all of the internal circuitry was taken out, then the potent-o-meter was taken out. Now the servo would perform like a dc motor. The dc motor is also able to run off of the eight-battery pack, which was the same battery pack used in the powering the microprocessor and sensor expansion pack. I also replaced all of the internal gears in each of the servos. After the hack was completed, the motors operated in the desired specifications as dc motors.

SENSORS

Scope

For Denise to meet the objectives of the fire, namely to find fire, she needed to use four different types of sensors. To move about freely without running into obstacles, Denise is equipped with micro-switches and infrared detectors. To tell when a fire was present, Denise has a smoke detector. To recognize fire, Denise has infrared detectors that detect wavelength of light that is emitted by a flame. By combining all of these sensors, my robot could be activated by fire and search for it

.

Micro-switches

Micro-switches were used in Denise to tell if she had bumped into any objects while moving about. The switches were built into the platform and were activated by a bumper that surrounded the robot's platform. Six switches were placed around the platform, and were grouped into two sets of three forming a front and rear bumper. The bumpers were connected to the sensor expansion board. When a bumper was hit an analog value was registered. By reading the analog value the microprocessor could distinguish which region the bumper was hit. In the design of my robot, I really didn't concern myself where the bump occurred only which bumper was hit. Programming for these sensors was quite easy, I would just check both bumpers for a hit, which would give an analog value greater than ten, and then react to it. The micro-switches used were the same kind used in Megatronics robots.

Infrared Emitters and Detectors

To really be effective in avoiding obstacles, Denise needed to be able to see object in front of her. To do this I used Sharp's infrared emitters and detectors. The emitters will emit a wavelength of 940 nm. The detectors are designed to detect the 940 nm wavelengths. Sharp's emitters and detectors are digital, meaning that when the detector sees an object in front of it, the signal it will return will be a logic one. By performing a hack provided by the MIL lab, I was able to turn the detectors from digital to analog. By use the detectors as analog, I can tell how far an object is away from the robot. After experimenting with the sensor, I got a range of 90 when there wasn't any object in front of it to 127 when an object was completely in front of the robot blocking the sensor.

After experimenting, I found that setting the infrared sensor's threshold to an analog value of 117. The reason I chose this as the threshold was because at 117 the object that the sensor is detecting is at 4 inches from the robot. This gives Denise enough distance to successfully avoid the obstacle.

Smoke Detector

The smoke detector that I am using is called an ionization alarm. The ionization alarm are generally more effective at detecting fast, flaming fires which consume combustible materials rapidly and spread quickly. These fires are usually papers burning in a wastebasket or a grease fire in the kitchen. The other type of smoke detector is called photoelectric alarm. These detectors are generally more effective at detecting slow, smoldering fires which smolder for hours before bursting into flames. The sources of these fires can include cigarettes burning in couches or bedding.

The reason that I chose the ionization alarm was that is the most cost effect sensor. If one would want to change and use to photoelectric alarm, the code I will be writing will be interchangeable. The circuitry is basically the same and the signal that I am using will be the same for each of the smoke detectors.

To use the smoke detector as a sensor, I needed to be able to detect a signal from the smoke detector that was useful. This signal needed to be less than five volts and greater than zero volts. After many attempts of testing different nodes when the alarm signal is on, I found that the voltage pulses to the speaker would be useful. When the smoke detector detects smoke, the speaker is set off. The voltage goes from zero volts to a voltage ranging from two to three volts. Even though the signal I will be using is analog, it does not change if the smoke source varies. The signal from the smoke detector will be fed into an analog port. When the port reads a voltage greater than 1.5 volts the robot will start a routine that searches for the heat source of the fire. The 68HC11 will be sampling to analog port constantly so that it will effectively be able to recognize when the smoke detector has detected smoke.

The smoke detector will be configured on my robot's platform as seen in figure two.

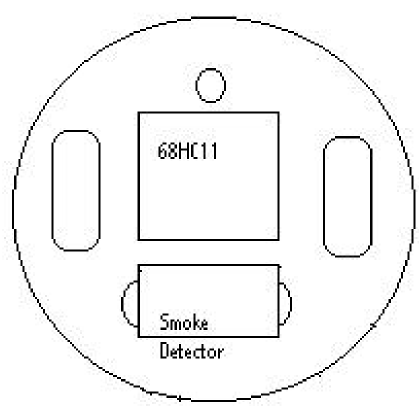


Figure Two.

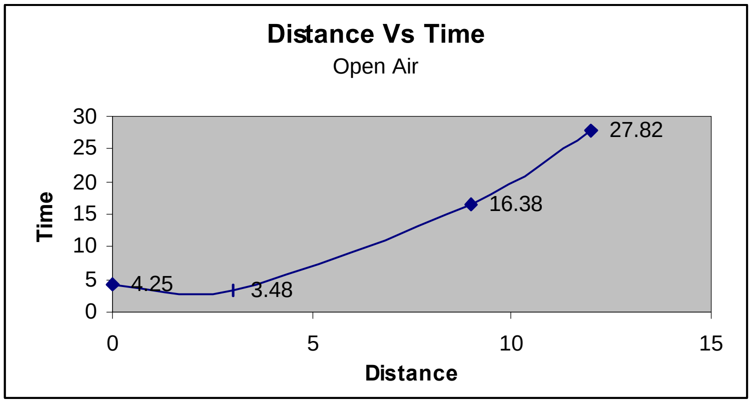
Experimental Data

I performed three different experiments to test the effectiveness of my smoke detector. The first experiment was test outside in the open air. This experiment is the most complicated and unrealistic test I could perform. First the smoke detector is not meant for outdoor use, but because I did not want set off the smoke detectors in the lab, which would cause the people in the lab to leave. This caused me to do all of my experiments outdoors. Because of this my testing was not vary accurate measuring the distance versus the time between when the smoke introduced to when the alarm detects smoke. The smoke was introduced by the use of one match being blown out. The second test was performed with the smoke detector placed at the end of a 14-inch cube box. I was able to get better readings for the distance versus time until the alarm detects smoke.   The third test measured the amount of smoke versus time of detection. This was performed lighting multiple matches at a standard length.

Data

Table one and graph one shows the data gathered by test one.

Distance Inches	Time seconds
0	5.24
0	4.56
0	2.95
Average	**4.25**
3	3.66
3	2.37
3	4.40
Average	**3.48**
	22.26
9	15.56
9	11.36
Average	**16.39**
12	30.25
12	31.94
12	21.26
Average	**27.82**

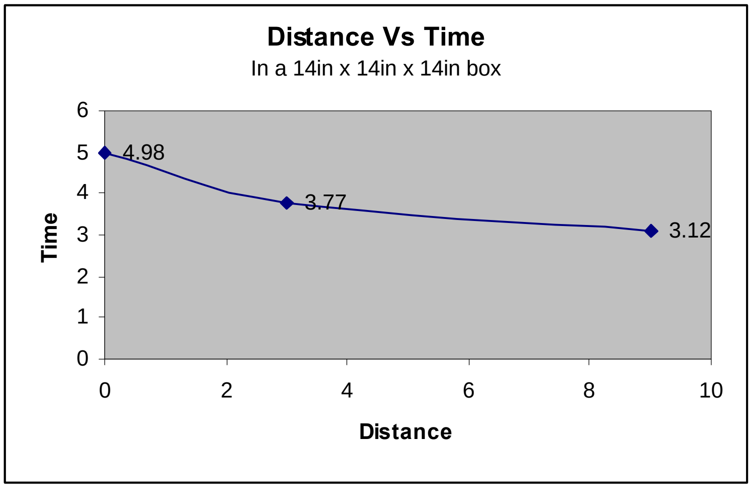


Graph One.

Table two and graph two shows the results of the second test.

Distance Inches	Time seconds
0	5.93
0	5.71
0	3.30
Average	**4.98**
3	3.51
3	3.19
3	4.62
Average	**3.77**
9	2.01
9	4.14
9	3.22
Average	**3.12**

Table Two

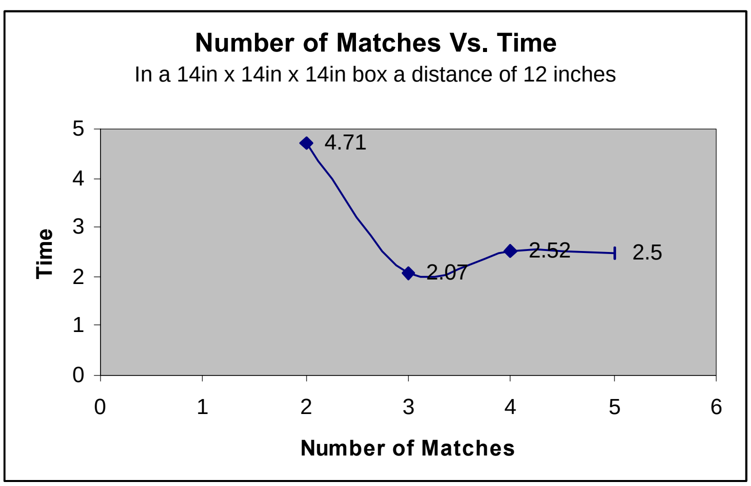


Graph Two

Table three and graph three shows the results of test three.

Number of Matches	time seconds
2	4.71
3	2.07
4	2.52
5	2.50

Table Three



Graph Three

Flame Detector (Infrared)

Once Denise was able to tell if there was a fire or not and wonder around a room safely, I needed Denise to be able to recognize if there is a flame. To accomplish this, I used an infrared detector. The detector detected light with the wavelength of 1000 nm. This is the same wavelength produced by a flame. The detector was unaffected by room light and the light emitted by the infrared diodes. The detectors were very accurate in detecting light. The detector, although, needed to be run through a amplifier (See Appendix) before the signal could be inputted into one of the analog port located on the sensor expansion pack. Even though the signal from the sensor was used as a analog signal it really acted like a digital signal. When a flame was in the range of the sensor, the signal received ranged between .76 volts and 4.9 volts. I can not explain why the signal differed in voltage reading, but after experimentation the signal strength was in no way related to the distance on the flame.

Problems

One problem I did notice was that the two sensors did not have the same range. After checking both of them I found out that the amplifiers were causing the problem. The reason for this problem was that the amplifier used two five mega-ohm resistors. The make the resistor I used two ten mega-ohm resistors in parallel. With the tolerances and the use of such large resistors, the resistors used in each amplifier were in the same. As it occurred, one of the detectors can ‘see’ a flame 6 inches away and the other can only ‘see’ a flame about two inches away. This still meets the specifications because that are used together to detect the flame.

BEHAVIORS

Bumper Sensing

Denise has the ability know if she has been bumped. She is able to tell if the bump occurred in the front or rear. The reaction to the bump is different depending on whether she was hit from the front or behind. If contact occurs from the front, Denise will back up at full speed and turn 180 degrees and proceed forward again. If the contact occurs from the rear, Denise first reduces her motors to 50 percent and then turns 90 degrees and proceeds forward at full speed again. Bump sensing is hopefully never used, but is a back up to the obstacle avoidance.

Obstacle Avoidance

Moving about freeing in a room is a very important behavior for Denise. She needs to be able to navigate through a room without hitting any objects. Denise uses her infrared sensors it tell how for she is away from obstacles that are in her way. When an object is about four inches away from Denise, she will start to react. If the object is to the left of Denise, she will turn to the right until she does not detect the object any longer. If the object is to her right, then she will turn left until the object is no longer in her view.

Smoke Detecting

When Denise is first started, she does nothing until see senses that there is smoke. Denise is constantly poling the smoke detector sensor, checking to see if there is smoke. Once smoke is detected, Denise starts to move about the room looking for the flame.

Flame Detection

After Denise starts to navigate throughout the room, she is also looking for a flame source. Once Denise has found the source, she then notifies people by turning in circles until you reset her. When the flame detector 'see' a flame, that is when Denise starts to signal that she had found the fire.

EXPERIMENTAL LAYOUT AND RESULTS

After completing the building of the platform and mounting all of the sensors, I started test Denise. First, to tell where the sensors were I used the program provided by the MIL lab for the testing of the Talrik's sensors. After I was able to pin point where the sensors I was used were located, I wrote a program to test my robot. Denise did not move, but printed out to the screen what she was doing. This form of debugging was very helpful because if the correct message was not getting display to the screen I knew that there was a problem. After debugging that program, I added the motor commands. I still kept the print statements, because it was still easier to debug.

After all of the debugging, I removed all of the print statements and started test Denise in the real environment. I first noticed that my program practice was very good at debugging because I really didn't need to do anymore programming. Denise was displaying the entire set of behaviors correctly. It was quite exciting to see Denise avoiding obstacles and finding fire. During my tests, I did not test the smoke detector every time. I used the test switch because the smell of the matches was starting to make me sick. In the end, Denise was performing the demo for demo-day without fail.

CONCLUSION

Building this robot was a great way of combining all of the knowledge that I have learned over the four year at Florida. I was able to create a project and build it on my own. I learned how to manage my time when things were slow. Building the robot taught me how to accomplish a goal. I did have some regrets though. I really wanted to have a extinguishing system that used CO_2 to extinguish the flame. Do to the lack of time, I was unable to add this to Denise. I also had build, but not working, a heat following behavior. The heat following used to temperature sensors that were able the measure the temperature in 10m volts per degree F. The behavior was going to help Denise be able to find the fire faster. The work that was done on this robot was very helpful in discovering what electrical engineering really is.

DOCUMENTATION

Megatronic. www.megatronic.com. Infrared emitters and detector; servos, 68HC11, sensor expansion pack and information on the servo hack and infrared hack.

IMDL. www.mil.ufl.edu/IMDL. Information on the process of the design and building of a robot

National Semiconductor. www.nsc.com. Temperature sensor (LM34-C)

Elctronics Plus. 34th Street Plaza. Infrared detectors and amplifying circuit.

Radio Shack. Resistors and circuit boards.

APPENDIX

Amplification Circuit

The following circuit in figure three was used to amplify the signal of the infrared detector. By amplifying the signal, the 68HC11 was able to use the sensor to detect a flame.

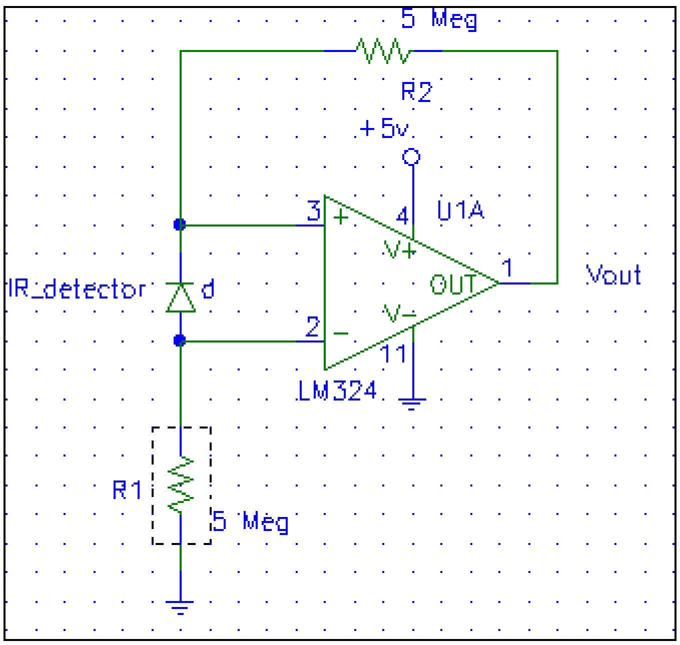


Figure Three.

Code for Denise

The following program was used to control Denise's behaviors.

```
/**********************************************************************
*
 * Title        smokeden.c
 * ProgrammerSean Justice
 * Date         July 17,1999
 * Version      1
 *
 * Description
 * Start by hit the rear bumper and then Denise will not do
 * anything until she has detected smoke.
 *
 *        Functions:
 *                    detectsmoke() used to detect if the smoke detector
 *                    has detected any smoke
 *
 * Use Hyperterm with the following settings:
 *
 *      COM1, 9600 Baud, 8 bits, No Parity, 1 Stop bit, No Flow
 *      of Control, VT100 Terminal, WrapLines.
 *
```

```
 *Usage: Verify functionality of TALRIK's sensors.

************************************************************************
/

/*************************** Includes **********************************/

#include <tkbase.h>



/*********************** End of includes ******************************/



/************************** Constants **********************************/

/*IR emitter output port driver, the output latch at address 0xffb9 */
#define IRE_OUT   *(unsigned char *)(0xffb9)

/*Constant for driving all the 40KHz modulated IR emitters on when loaded into
IRE_OUT */
#define IRE_ALL_ON 0xff
```

```
/*Constant for turning all the 40KHz modulated IR emitters off when loaded into
IRE_OUT */
#define IRE_ALL_OFF 0x00

#define BUMPER_FUZZY 10
#define NOSMOKE 40
#define IRTHRESHOLD 110
#define FIRE 45
#define FIRE2 40
#define HEAT 45

/************************ End of Constants ******************************/
int rspeed = 0;
int lspeed = 0;
unsigned int temp_one = 0;
unsigned int temp_two = 0;
unsigned int left = 0;
unsigned int right = 0;
/******************functions***********************************/

void detectsmoke(void);
void avoidobject(void);
void righturn(void);
```

```
void lefturn(void);

void checkbumper(void);

void firecheck(void);

void intitemp(void);

void lookright(void);

void lookleft(void);

void heatcheck(void);

void motorforward(void);

void heatsource(void);

void heatforward(void);

void firesignal(void);




void main(void)

/**************************** Main *********************************/

{

  init_analog();

  init_motortk();

  init_clocktk();

  init_serial();


  IRE_OUT = IRE_ALL_ON;    /* Turns on all the 40KHz modulated IR emitters */
```

```
wait(300);

while (rear_bumper() < BUMPER_FUZZY); /*To start press the rear bumper*/

read_IR();
wait(35);

detectsmoke(); /*don't do anything unitl smoke is detected*/
intitemp();     /*See what the temperature in the room is*/



while (1)
{
        checkbumper();          /*See if bumper has been hit*/

        firecheck();     /*checks if there is a flame*/

        lookright();     /*look to the right*/
        lookleft();                    /*look to the left*/
        motorforward();
        heatcheck();    /*check for heat increase*/
```

```
  }
}
/**********************fucntions*************************************
*********/

/* function to detect the smoke, this function will alert Denise to the problem*/
void detectsmoke(void)
{
	read_IR();
	wait(35);

	while (IRDT[IR5clk] < NOSMOKE)
		{
			read_IR();

			wait(35);
		}

}
/* function initize the heat sensors for the temp of the room */

void intitemp(void)
{
```

```
	read_IR();

	wait(35);


	temp_one = IRDT[2];

	temp_two = IRDT[3];


}

/* this program checks the bumper front and rear to see if anything has hit it*/


void checkbumper(void)

{

	if (rear_bumper() > BUMPER_FUZZY)

		{


			rspeed = lspeed = 50;

			motortk(RIGHT_MOTOR, rspeed);

			motortk(LEFT_MOTOR, lspeed);

			wait(1000);

			rspeed = 100;

			lspeed = -100;

			motortk(RIGHT_MOTOR, rspeed);

			motortk(LEFT_MOTOR, lspeed);
```

```
				wait(1000);
			}
		else if (front_bumper() > BUMPER_FUZZY)
			{

				rspeed = lspeed = -50;
				motortk(RIGHT_MOTOR, rspeed);
				motortk(LEFT_MOTOR, lspeed);
				wait(1000);
				rspeed = -100;
				lspeed = 100;
				motortk(RIGHT_MOTOR, rspeed);
				motortk(LEFT_MOTOR, lspeed);
				wait(1000);

			}
}
/* this funciton checks to see if there is a fire in front of the robot*/


void firecheck(void)
{
		read_IR();
```

```
        wait(35);

        if ((IRDT[IR6clk] > FIRE) || (IRDT[IR7clk] > FIRE2))
        {

                rspeed = -100;
                lspeed = 100;
                motortk(RIGHT_MOTOR, rspeed);
                motortk(LEFT_MOTOR, lspeed);
                while (1);
                /*firesignal();*/

        }
}
/*this functions looks to the right to see if there is an object*/
void lookright(void)
{
        read_IR();
        wait(35);

        if ((IRDT[IR4clk] > IRTHRESHOLD) || (IRDT[IR3clk] > IRTHRESHOLD))
```

```
                {
                        left = 1;
                        lefturn();

                }
}
void lookleft(void)
{

        read_IR();
        wait(35);


         if ((IRDT[IR1clk] > IRTHRESHOLD) || (IRDT[IR2clk] > IRTHRESHOLD))
                {
                        right = 1;
                        righturn();
                }

}
void righturn(void)
{
        if (right == 1)
```

```
	{
		rspeed = -100;
		lspeed = 100;
		motortk(RIGHT_MOTOR, rspeed);
		motortk(LEFT_MOTOR, lspeed);
		while ((IRDT[IR1clk] > IRTHRESHOLD) || (IRDT[IR2clk] >
IRTHRESHOLD))
		{
			read_IR();
			wait(35);

			firecheck();

		}
	wait(150);

	}
}
void lefturn(void)
{
	if (left == 1)
	{
		rspeed = 100;
```

```
			lspeed = -100;
			motortk(RIGHT_MOTOR, rspeed);
			motortk(LEFT_MOTOR, lspeed);
			while ((IRDT[IR3clk] > IRTHRESHOLD) || (IRDT[IR4clk] >
IRTHRESHOLD))
			{
				read_IR();
				wait(35);
				firecheck();



			}
		wait(150);
		}
}
void motorforward(void)
{
		rspeed = 100;
		lspeed = 50;
		motortk(RIGHT_MOTOR, rspeed);
		motortk(LEFT_MOTOR, lspeed);
}
void heatcheck(void)
```

```
{
	read_IR();
	wait(35);

	while ((IRDT[2] > HEAT) || (IRDT[3] > HEAT))
	{
		read_IR();
		wait(35);
		lookright();
		lookleft();
		rspeed = -60;
		lspeed = 30;
		motortk(RIGHT_MOTOR, rspeed);
		motortk(LEFT_MOTOR, lspeed);

		heatsource();

	}
}
void heatsource(void)
{
	int search = 0;
```

```
read_IR();
wait(35);

temp_one = IRDT[2];
temp_two = IRDT[3];

rspeed = 60;
lspeed = 30;
motortk(RIGHT_MOTOR, rspeed);
motortk(LEFT_MOTOR, lspeed);

while (search == 0)
{
	read_IR();
	wait(35);


	if ((IRDT[2] > (temp_one - 2)) || (IRDT[3] > (temp_two - 2)))
	{
		temp_one = IRDT[2];
		temp_two = IRDT[3];
		firecheck();
	}
```

```
        if ((IRDT[2] < (temp_one + 2)) || (IRDT[3] < (temp_two + 2)))
        {
                search = 1;
                temp_one = IRDT[2];
                temp_two = IRDT[3];
                rspeed = lspeed = 0;
                motortk(RIGHT_MOTOR, rspeed);
                motortk(LEFT_MOTOR, lspeed);
                firecheck();
        }
}

/*while (search == 1)
{
        read_IR();
        wait(35);

        rspeed = 40;
        lspeed = -40;
        motortk(RIGHT_MOTOR, rspeed);
        motortk(LEFT_MOTOR, lspeed);
        while ((IRDT[2] < (temp_one + 2)) || (IRDT[3] < (temp_two + 2)));
        search = 2;
```

```
		firecheck();
	}
	heatforward();*/
}
void heatforward(void)
{
	rspeed = lspeed = 50;
	motortk(RIGHT_MOTOR, rspeed);
	motortk(LEFT_MOTOR, lspeed);
	while ((IRDT[2] > HEAT) || (IRDT[3] > HEAT))
	{
		firecheck();
		lookright();
		lookleft();
	}
	heatsource();
}
void firesignal(void)
{
	rspeed = -100;
	lspeed = 100;
	motortk(RIGHT_MOTOR, rspeed);
	motortk(LEFT_MOTOR, lspeed);
```

```
        wait(1500);

        rspeed = 100;

        lspeed = -100;

        motortk(RIGHT_MOTOR, rspeed);

        motortk(LEFT_MOTOR, lspeed);

        wait(1500);

        while(1);

}
```