University of Florida

Department of Electrical Engineering

EEL 5934

Robotics

Written Report #3

Title: Building Security Robot

Prepared by: Chung-Kei Savio Woo
Instructor: Keith L. Doty

Date: 12/4/95

TABLE OF CONTENT

ABSTRACT

In big cities, the security guards of a building are facing many problems such as bombs, theft, and fire. If there is a robot that can replace humans in the management of this kind of problems, the dangers these jobs are exposed to will be reduced. Therefore, the idea of a security robot is popular because a robot can work 24 hours a day, 7 days a week, and the risk of accidents to human life is reduced. Also, in the transfer of information and instructions, a robot is more efficient because data transition to a robot via computer minimizes error.

In this project, the prototype of a security robot will be built for further large scale development. The security robot will be designed to perform in important situations in which it would be too dangerous for humans to be a part of. The robot will have Hall-effect sensors that can locate bombs before they are triggered. With the rechargeable battery function, it is possible for the robot to run 24 hours a day. Since it is equipped with a smoke detector, it can find any source of fire. With the prototype, problems with the design and concepts can be identified more easily. It will tell us what we should implement on the security robot and what we should not.

The main purposes of the security robot are to reduce the cost of hiring people, reduce the risk of accidents, and improve the efficiency of a security system. With the security robot, the bombing of the World Trade Center might have been avoided. It helps us take a step toward the future security system.

INTRODUCTION

Background

The security of the modern world is threatened by many problems such as bombings and theft. The current security system depends mainly on human support and no single person can work 24 hours a day. Therefore, a robot is very useful in the modern security system because it can fully function for 24 hours a day and 7 days a week with the support of a power recharge station. It can detect the source of smoke, which may indicate fire, and it can also detect bombs that may cause a tragedy. With all these problems, it is necessary to develop an auto-security system which leads to the idea of a security robot.

Scope and Objective

The Building Security Robot (BSR) project is designed to implement many different functions such as heat detection, collision avoidance, sound effect, and light-following (optional functions are self-recharging, bar code reader, and IR communication). The objective of the project is to construct a building security robot with MC68HC11 board on circular wood platform with three wheels. In building the prototype of the security robot with the EVBU board, we will understand the problems that may occur in a larger scale project to build a human size security robot. With the Building Security Robot (BSR), the cost of a security system will be reduced due to the hiring of fewer people, the reduction in injuries due to tragedies or accidents. Also, the efficiency of the security system will be improved. The goal of this project is to provide a small scale BSR that will assist the further design and development of the large scale BSR.

Structure of the Report

In the remaining sections of the report, the initial installation of BSR will be discussed in more detail. The following sections will concentrate on a specific catalog after discussion of the entire integrated system in section 3. The platform, actuation, and sensors will be discussed in section 4, 5, and 6, respectively. The current behavior and some Experimental Results will be mentioned in section 7 and 8, respectively. The scope, specifications, and objectives of each catalog, as well as the theory of operation of each area, will be explained.

INTEGRATED SYSTEM

Overview

The Building Security Robot is named “Bruce” in memory of Bruce Lee. The BSR is going to be built on a 68HC11 with 32k SRAM. The other hardware are mainly sensors that allow BSR to fulfill all the intended functions. The software provides algorithm (IC), which controls the motion of the robot and analyzes the data from the sensors. Currently, the IR sensors and IR LEDs are implemented on Bruce to fulfill the function of collision avoidance. The CdS cells will be used for light-following behavior, and the pyroelectric sensor will be used for the heat motion sensing behavior. The software is still in process and will be ready during the demonstration time. The entire integrated system follows the specifications given below.

Range: Room size

Main Control: MC68HC11 with 32k SRAM

Power: 8 AA Alkaline or AA Nickel-Cadmium battery (12V)

Output Latch: 8-bit 40k Hz signal with 0V to 5V range

Backup: Backup power for SRAM

Motor: Futaba servo motor (FP-S148)

IR sensors: Sharp GP1U52X

Organizational Description

There are several different features that will be implemented on Bruce. Currently, Bruce has a circular wood platform of 10 inches in diameter and 3.3 inches in height above the ground. It is driven by two servo motors with one caster wheel at the back. Bruce has reduced to four IR sensors and four IR LEDs with simple algorithm that allow Bruce to avoid obstacles. The four IR sensors are connected to the input of an analog multiplexer (MC14051), and the output of the MC14051 is connected to PE0 pin (A/D). The four IR LEDs connected to pin 1 to pin 4 of the output latch with 40k Hz signal. Three CdS cells are connected to another MC14051 which output is hooked up to PE1 pin (A/D). The output of the pyroelectric sensor circuit is directly connected to the PE2 pin (A/D). Bruce has three basic behaviors such as collision avoidance, light-following, and heat motion sensing.

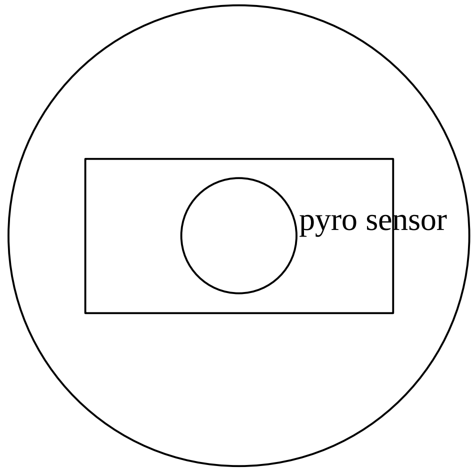
CIRCULAR WOOD PLATFORM

Objective

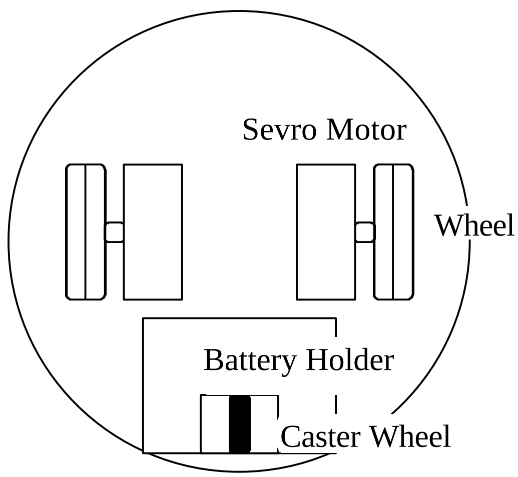
The security robot of a building must be able to walk through long, narrow passages of each floor of the building. Therefore, the robot must have a straight line function and a wall follow function. According to the text, Mobile Robots, between a round robot and a square robot that have the same width and a simple turn-while-in-contact algorithm, the round robot will negotiate a long and narrow passage more efficiently. The square robot will need more movement.

Description

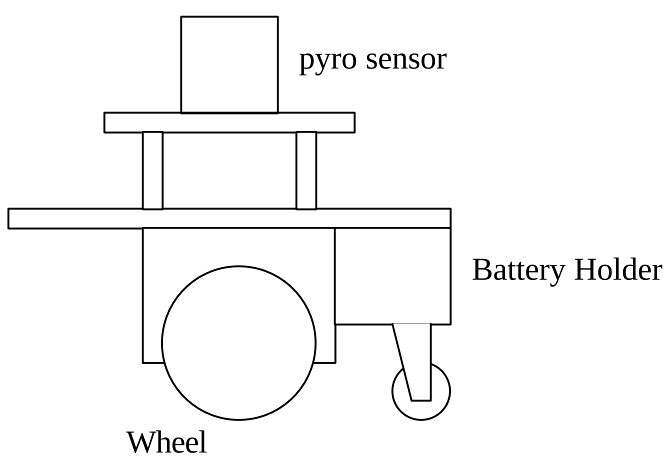
On Bruce, there is a round wood, double level platform with 10 inches diameter. With an additional level, it provides more space for the pryoelectric sensor. The 68HC11 is placed on the first level of the wood platform with screws and nuts. A battery holder is implemented in the back of the bottom of the wood platform, and the caster wheel is glued on to the bottom of the battery holder. The weight of the battery helps to balance the robot so it will not fall to the front. The servo motors are placed on the wood log that is attached directly to the bottom of the wood platform. The following, Figure 1, is a sketch of the wood platform with wheels.



Top View of Bruce



Bottom View of Bruce



Side View of Bruce

Figure 1: Top, Bottom, and Side View of Bruce

ACTUATION BY SERVO MOTORS

Objective

On a Building Security Robot, the efficiency and quality of the motor are very important. With better quality motors, there will be less chance for the robot to be held by an obstacle and lose all battery power when trying to disengage from the obstacle. There is better speed control and torque control with better quality motors.

Description

Bruce is designed to use Swiss motors for the wheel driver, and a Swiss motor is the best quality motor that I can find in comparison to a servo motor. However, I have spent one month to design and build the frame for the Swiss motor, and I was not able to finish the job. In order to save more time on the design on sensors, I gave up the Swiss motor design and used the current servo motors. The reason of using servo motor is the Futaba servo motor is the cheapest motor we can get with high torque gear feature. Using lib_rw11.c instead of lib_rw10.c, it gives more levels for speed control on servo motors because lib_rw11.c use interrupt instead of I/O. Therefore the servo motor is efficient enough for speed control in collision avoidance and shaft encoder. The following are the specifications of the Futaba servo motor (rugged low-profile servo).

Power supply	: 4.8V to 6.0V
Power consumption	: 6.0V 8mA (at idle)
Output torque	: 42 oz/in
Operation speed	: 1.32 sec/rev
Dimensions	: 1.59 x 0.77 x 1.4 in

Weight : 1.5 oz

In order to control the servo motors, we used L293 motor driver chip which available in the IMDL. The schematic of the motor driver is given in the memory expansion handout. The direction of the motors is controlled by the PD4 and PD5 pins (Port D), and the speed of the motors is controlled by the PA5 and PA6 pins (Output Capture) using pulse-width to modulate the left and right enable pins.

IR SENSOR AND IR LED

Objective

The Building Security Robot will need sensors to perform features such as collision avoidance, heat detection, and metal detection. Currently, collision avoidance is needed to avoid obstacles and to do the wall follow function. Therefore, the position of the sensors is very important due to the implementation of the software.

Description

Previously, there are four IR sensors (Sharp GP1U52X) implemented on the bottom of Bruce, two cross IR in the front and two IR on the side (one on each side). The change has been made such that there are two cross IR in the front and two straight-looking IR on the left and right. The shield of the sensors is grounded and only analog output is used only. The IR LED is 1.02 V each and the power consumption varies with different resistor values (current resistor value is 820 ohms). There is a sketch in Figure 3 that shows the bottom look of the implementation of the IR sensors and IR LEDs. The four IR sensors are connected to an analog multiplexer which output is hooked up to PE0, and the four IR LEDs are connected to the output latch pin 1 to pin 4. The Figure 4 on next page shows the control of the IR sensors. The sensor software is not fully developed, and I hope it will be ready on the demonstration. The sensors are tested individually with different resistor values in order to know sensitive of the IR sensors on robot in different environment (blue carpet and white carpet).

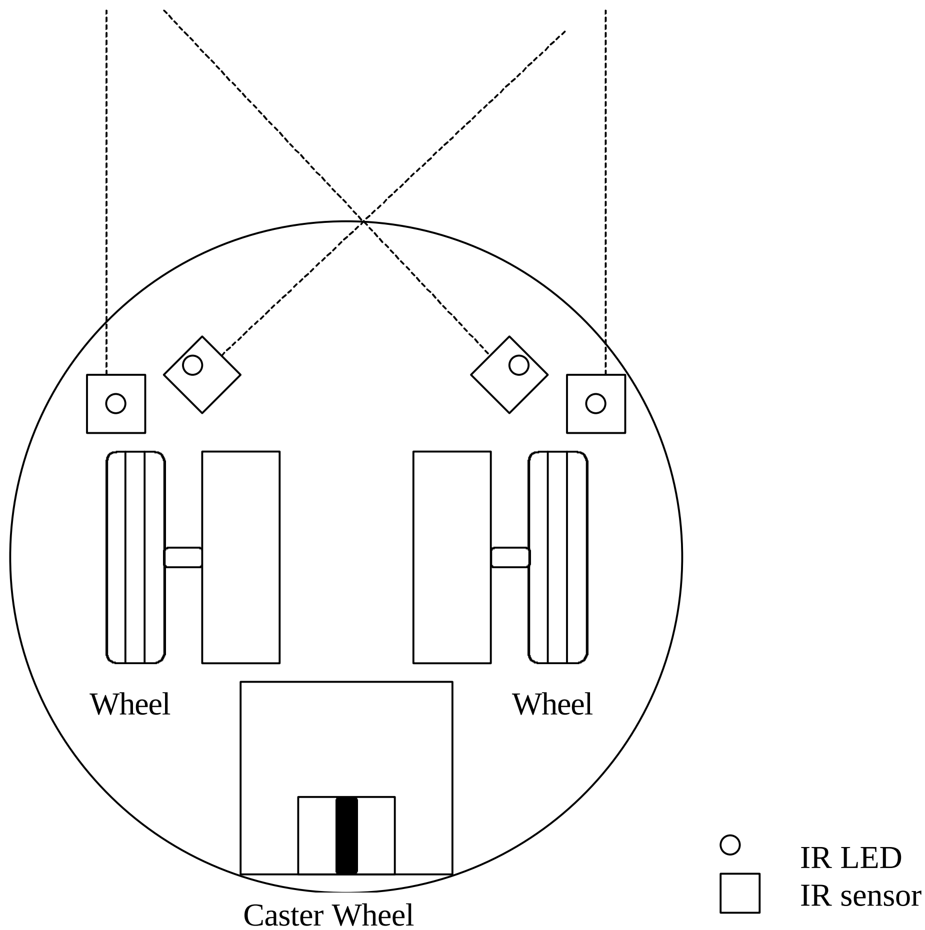


Figure 3: The Bottom View of the Robot

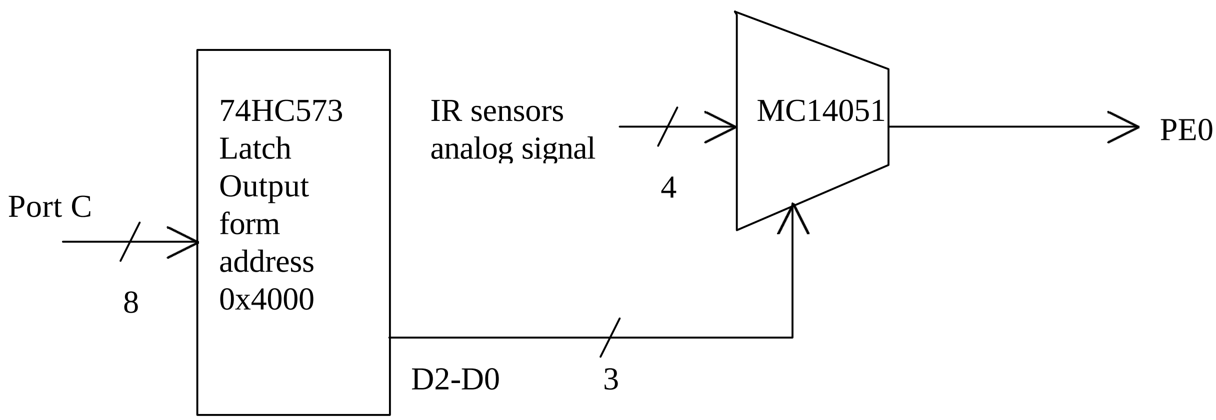


Figure 4: The Control of IR Sensors

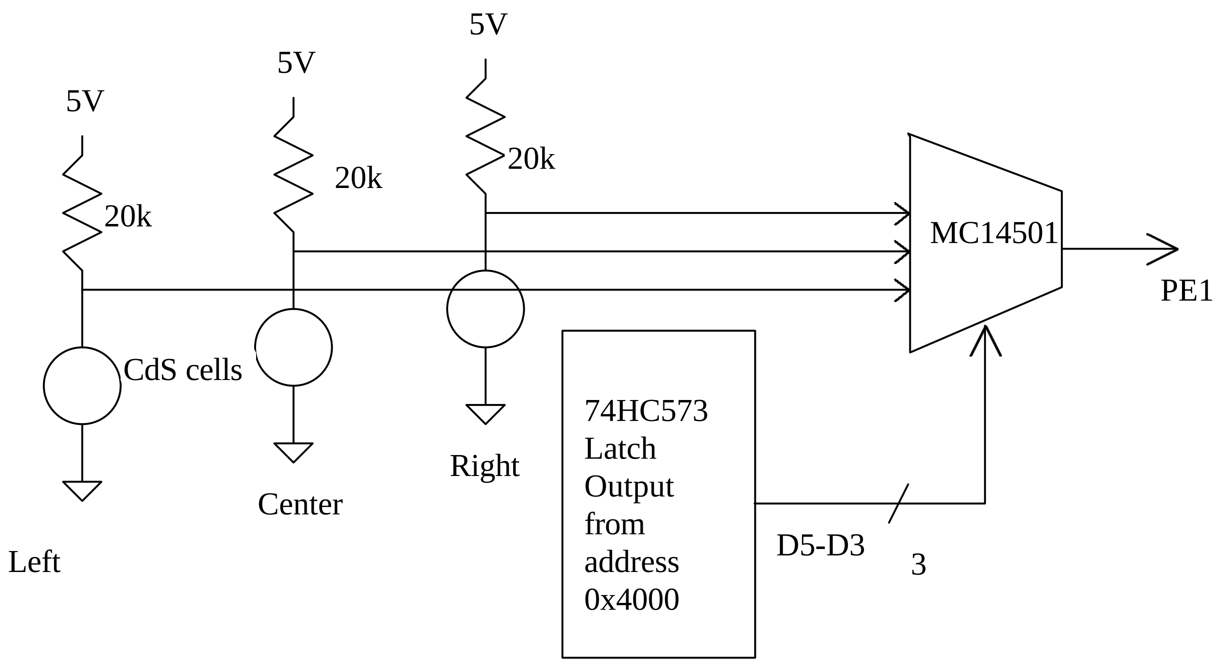
CADMIUM SULFIDE CELLS

<u>Objective</u>

The Building Security Robot will need to work at dark area, and there may be an intruder if there is an unknown light source. Light-following behavior will allow the robot to follow and identify the light.

<u>Description</u>

Cadmium Sulfide (CdS) photocells is a special type of resistor which responds to light. The more light hitting the photocell, the lower its the resistance. According to the diagram below, the output signal of the photocell is an analog voltage corresponding to the amount of light hitting the cell. Higher values correspond to less light. I shielded the CdS cells with the heat shrink tube in order to restrict the amount of light striking the sensor to the direction I expect the light to be coming from. The highest digital value obtained from the A/D pin is 255 (5 V) in a totally darkness environment, and the lowest digital value is 4 which I pointed a torch directly to the photocell. The CdS cells are used to implement the photosensitive “eye” which is an array of three CdS cells. For the eye, the array of CdS cells is aimed parallel with the ground. The diagram on next page shows the circuit diagram of the implementation of CdS cells as eye.


5V
5V
5V
20k
20k
20k
MC14501
PE1
CdS cells
74HC573
Latch
Output
from
address
0x4000
Right
Center
Left
D5-D3
3

PYROELECTRIC SENSOR

Objective

The Building Security Robot will need to detect the unknown source of heat such as body temperature of intruder or a fire source. The pyroelectric sensor is used to detect changes in temperature of the sensor, it can be used to implement the heat-sensing behavior.

Description

The pyroelectric sensor is one of the most useful sensors for endowing the robot with a means of interacting with humans. The pyroelectric sensor is one of the component inside the motion-detector for light of garage door. The output of a pyroelectric sensor changes when small changes in the temperature of the sensor occur over time. There are two pieces of lithium tantalate crystals, and charge is induced as the crystal is heated. When there is a heat source passes in front of the sensor, there is a voltage change from 0V to 5V after the pyroelectric sensor signal has been amplified. I connected the output of amplified signal to the PE2 pin, and the motion detector is implement on the top level of the robot. When the robot moves around according to its own vertical axis, the heat body can be detected and chased.

COLLISION AVOIDANCE BEHAVIOR

Objective

With the Building Security Robot, the robot is implemented with all the sensors that allow for the behavior of collision avoidance and wall follow function. The software will be used to control the behavior, and the algorithm must be detailed enough to not cause the robot to hit the wall or be held by an obstacle.

Description

On Bruce, the robot has two cross IR sensors on the front that detect the distance between the obstacle and the front of the robot. The algorithm will the robot to turn until the reading of the IR below the threshold. The two straight-looking IR will used to detect the obstacle that the two cross can't see. Therefore, the robot will maintain a distance and not hit the wall. The algorithm of collision avoidance is not yet finished, and there is pseudocode in the Appendix A which help to describe the behavior. The step speed control is used to control the speed of motors which not to go fully forward and then fully backward, and it increment or decrement motor speed in 5 % in a time.

LIGHT-FOLLOWING BEHAVIOR

Objective

With the photosensitive “eye” which implemented by using CdS cells, the robot can search for light source. With simple algorithm and simple circuitry, CdS cells can work in different environment.

Description

With the simple algorithm in the example robot Critter, Bruce can follow light. For light-following behavior, the array of CdS cells will input analog signal to A/D and the robot will always turn to the side with the lower voltage reading and turn away from the higher voltage side. The code has not finish yet, and there is pseudocode in the Appendix A which help to describe the behavior.

HEAT MOTION SENSING BEHAVIOR

Objective

With the motion detector which contains an essential component, pyroelectric sensor, the robot can following heat source. The pyroelectric sensor can detect the infrared energy emitted by human; therefore, the intruder can be chased.

Description

The motion detector is implemented on the top level of the robot, and it can sense a tall heat object. The robot will initially turn around to look for heat source. If it cannot find a heat source, it will move forward for 5 second. Then it will turn around again to look for heat source, and it become a loop of the behavior. If it find a heat source, the robot will stop after it read the signal and move forward for 1 or 2 seconds. Then it turn around in the opposite direction to look for the same heat source. After the sensor passed the heat source again, it will stop again and move forward for 1 or 2 seconds. The algorithm will make the robot to chase the source forever.

EXPERIMENTAL LAYOUT AND RESULTS

Motor Testing

With the motor implemented, we used the IC command such as motor(0,100); and motor(1,100); to test the function of the motor. With the percentage of the input such as 100 and -100, the motor is able to turn forward and backward. There are only 7 levels if the percentage changes, which means there are only 7 speed levels due to the motor(); command. With the lib_rw11.c, I found out there are more speed level and float-point speed control. The testing of the servo motors are done in the lab.

Pyroelectric Sensor Testing

With the motion detector, I found the output of the amplifier which I think that is the amplified signal of pyroelectric sensor. Then I connected that pin to the scope to view the signal. With the heat body passes in front of the motion detector, the scope indicated that there is a 0V to 5V swing.

CdS Cell Testing

In the experiment, I found out if the pull-up resistor value is equal to 47k ohm, the highest power consumption is 5V 0.1mA, the range from bright to half bright is shorter. If the resistor value is equal to 20k, the power consumption is higher and the range from bright to half bright is larger. The following table shows the data of the experiment.

CdS Cell	Dark	Dimmer Light	Flash light pointed center	Flash light pointed right	Flash light pointed left
	Port E value				
center	254	182	12	132	150
left	254	202	110	16	198
right	255	215	155	238	11

CONCLUSION

In this Building Security Robot project, the purpose is to build the prototype of the BSR which will identify potential problems and help the further development and design of the large scale BSR. The main idea of building Bruce is to reduce the cost of hiring the extra people, reduce the risk of accidents, and improve the efficiency of a security system. One limitation of the work to be noted is that the project may be too small in scale to realize the problems of the large scale project. For future work, many more sensors may be needed for more features. For example, the coil will be used for metal detection.

REFERENCE

[1] Joseph L. Jones and Anita M. Flynn. Mobile Robots: Inspiration to Implementation

K. Peters 1993

[2] Fred Martin. The 6.270 Robot Builder's Guide

Epistomology and Learning Group, MIT 1992

[3] M68HC11EVBU Universal Evaluation Board User's Manual

Motorola Inc. 1992

[4] Fast and LS TTL Data

Motorola Inc. 1992

APPENDIX A

```
/* Name: Chung-Kei Savio Woo
   Robot: Bruce
   File: bruce.c  */

/*Variables*/
int left_eye, center_eye, right_eye;
float lspeedl, lspeedr;
int light_flag;
int lcount;

int pyro1;
int left = 0;
int right = 1;
float hspeedl;
float hspeedr;
int heat_flag;
int hcount;

int leftc, rightc, lco, rco;
float lcd, rcd, abslcd, absrcd;
int lefts, rights, lso, rso;
float lsd, rsd, abslsd, absrsd;
float ca = 20.0;
float cb = 10.0;
float aspeedl = 80.0;
float aspeedr = 80.0;
int avoid_flag = 1;

float motorl_output, motorr_output;

void avoid()
{
   /*Initial reading of IR sensors*/

  poke(0x7000,0x01);
  wait(100);
  leftc = analog(2);
  poke(0x7000,0x02);
  wait(100);
  rightc = analog(3)*2;
  poke(0x7000,0x04);
  wait(100);
  lefts = analog(4);
  poke(0x7000,0x08);
```

```
  wait(100);
  rights = analog(5);
  poke(0x7000,0x00);

/*Control loop*/

  while(1)
  {
  lco = leftc;
  rco = rightc;
  lso = lefts;
  rso = rights;

  /*Read IR sensors*/
  poke(0x7000,0x01);
  wait(100);
  leftc = analog(2);
  poke(0x7000,0x02);
  wait(100);
  rightc = analog(3)*2;
  poke(0x7000,0x04);
  wait(100);
  lefts = analog(4);
  poke(0x7000,0x08);
  wait(100);
  rights = analog(5);
  poke(0x7000,0x00);

  lcd = (float)(lco-leftc);
  rcd = (float)(rco-rightc);
  lsd = (float)(lso-lefts);
  rsd = (float)(rso-rights);

  abslcd  = lcd;
  if(lcd < 0.0) abslcd  = -lcd;
  absrcd  = rcd;
  if(rcd < 0.0) absrcd  = -rcd;
  abslsd  = lsd;
  if(lsd < 0.0) abslsd  = -lsd;
  abslcd  = rsd;
  if(rsd < 0.0) absrsd  = -rsd;

  if (abslcd >= 3.0 && absrcd >= 3.0 && abslsd >= 3.0 && absrsd >= 3.0)
    avoid_flag = 0;
  else
    avoid_flag = 1;
```

```
  aspeedl = aspeedl+cb*rsd+ca*lcd;
  aspeedr = aspeedr+cb*lsd+ca*rcd;

  if(aspeedr > 100.0)
  aspeedr = 100.0;
  if(aspeedl > 100.0)
  aspeedl = 100.0;
  if(aspeedr < -30.0)
  aspeedr = -30.0;
  if(aspeedl < -30.0)
  aspeedl = -30.0;


  }/*end of while*/
}/*end of avoid*/


void wait (int milli_seconds)
{
  long timer_a;

  timer_a = mseconds() + (long) milli_seconds;
  while (timer_a > mseconds()) {
    defer();
  }
}

void light_sensor()
{
  while(1) {
    poke(0x4000,0b00000000);
    center_eye = analog(1);
    poke(0x4000,0b00001000);
    left_eye = analog(1);
    poke(0x4000,0b00010000);
    right_eye = analog(1);
    wait(100);
  }
}

void light()
{
  while(1) {
    if (left_eye < center_eye && left_eye < right_eye) {
      lspeedl = 0.0;
```

```
      lspeedr = 80.0;
      tone(700.0,0.5);
    }
    else if (right_eye < center_eye && right_eye < left_eye) {
      lspeedl = 80.0;
      lspeedr = 0.0;
      tone(400.0,0.5);
    }
    else if (center_eye < right_eye && center_eye < left_eye) {
      lspeedl = 80.0;
      lspeedr = 80.0;
      tone(1000.0,0.5);
    }
    else {
      lspeedl = 80.0;
      lspeedr = -80.0;
      tone(2000.0,0.5);
    }
  }
}

void pyro_sensor()
{
  while(1) {
    pyro1 = analog(6);
  }
}

void heat()
{
  while(1) {
    while(pyro1 < 40) {
      hspeedl = -30.0;
      hspeedr = 30.0;
    }
    tone(800.0,0.5);
    hspeedl = 50.0;
    hspeedr = 70.0;
    wait(1500);
    while(pyro1 < 40) {
      hspeedl = 30.0;
      hspeedr = -30.0;
    }
    tone(800.0,0.5);
    hspeedl = 70.0;
    hspeedr = 50.0;
```

```
    wait(1500);
  }
}

void main()
{
  start_process(light_sensor());
  start_process(pyro_sensor());
  start_process(avoid());
  start_process(heat());
  start_process(light());
  heat_flag = 0;
  light_flag = 1;

  while(1){
  if(avoid_flag == 1) {
    motor(left,aspeedl);
    motor(right,aspeedr);
  }
  else{
    if(heat_flag = 1 && light_flag = 0) {
      while (lcount < 20000){
        motor(left,lspeedl);
        motor(right,lspeedr);
        lcount++;
        if(avoid_flag == 1) break;
      }
      lcount = 0;
      heat_flag = 0;
      light_flag = 1;
    }
    else{
      while (hcount < 20000){
        motor(left,hspeedl);
        motor(right,hspeedr);
        hcount++;
        if(avoid_flag == 1) break;
      }
      hcount = 0;
      heat_flag = 1;
      light_flag = 0;
    }
  }
  }
}
```