580.471 Principles of Design of Biomedical Instrumentation Challenge Project Report

The RainCane

Joseph Heng
Steve Wang

Introduction

Although the main objective of the project was to develop a device that provided sufficient feedback for a blind person to navigate through an obstacle course, we decided to develop a device that would be useful in two modes – an indoor mode, and an outdoor mode. This device, known as the RainCane, is essentially a modified umbrella outfitted with electronic sensors that provide sensory feedback to the user. Given time constraints and our choice of the outdoor competition, we focused on developing the outdoors mode of the RainCane, while maintaining a simplistic indoors mode, thus leaving room for improvements and further design for the indoors mode.

Design Process

Needs Identification

In coming up with a design solution to the problem of designing a smart cane for the blind for outdoors use, we had to identify several needs for the user and brainstorm for improvements over the current cane that most blind people use.

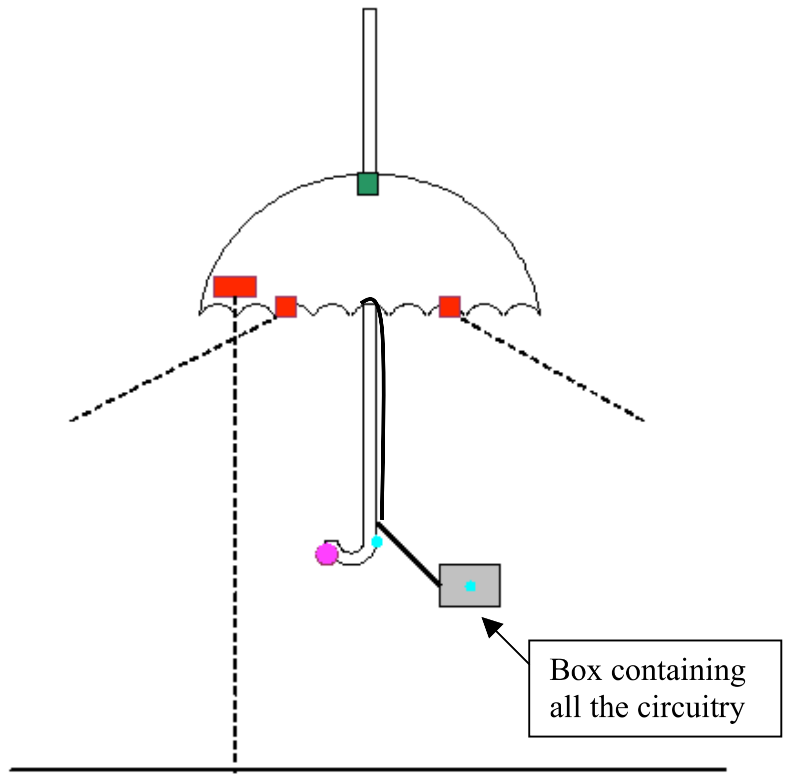
Several needs we identified that had to be addressed in the design of an outdoors smart cane were:

a) Detection of obstacles in the forward direction
b) Detection of people around the user
c) Detection of steps
d) An ability to guide the user towards a building
e) Weatherproof cane
f) Ability to use the cane indoors

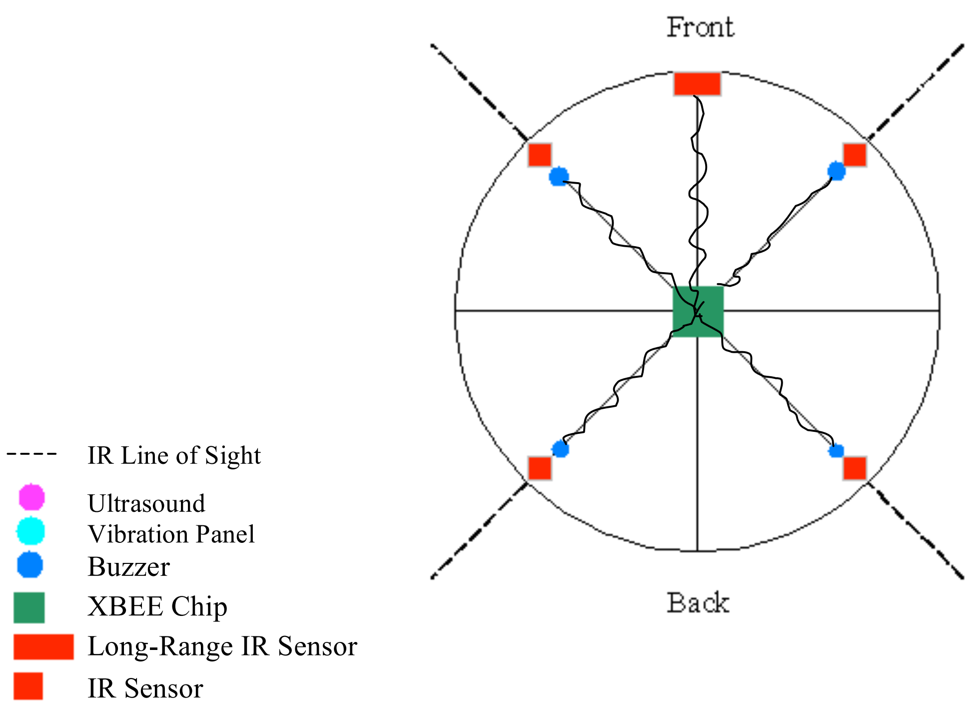
Design Solution - Sensor Selection, Microcontroller & Feedback

In coming up with a design solution, we settled on retrofitting a long umbrella with sensors that would interface with a microcontroller, which in turn would work with feedback devices such as vibration panels and speakers to provide the user with sensory feedback. For our device, the microcontroller selected was the PIC16F877A. It was selected primarily because of its many ports available, due to the many sensors and feedback devices we were using, and also because of our familiarity with it from previous labs that employed the microcontroller. These sensors and feedback devices are implemented in a design that is shown below:

SIDE


Box containing
all the circuitry

TOP


Front
Back
IR Line of Sight
Ultrasound
Vibration Panel
Buzzer
XBEE Chip
Long-Range IR Sensor
IR Sensor

Need (a) - Solution

To address need (a), the detection of obstacles in the forward direction, we selected an ultrasound sensor, the Devantech SRF04, which has a range of about 3 meters. This ultrasound sensor, placed on the front “hook” of the umbrella, would interface with a vibration panel, which was embedded in the handle of the umbrella. Pulse width modulation was used, to ensure that the closer the obstacle, the faster the panel would vibrate.

Need (b) - Solution

In addressing need (b), the detection of people around the user, we came up with a novel solution to place 4 infrared GP2D12 sensors on the crown of the umbrella. These sensors have a detection range of 3 feet. The 2 sensors at the front of the umbrella are placed closer to each other to have a greater probability of detecting people who walk along a collision path with the user. The 2 sensors at the back of the umbrella are spaced apart from each other at about 90 degrees. These 4 sensors are coupled with their respective high frequency buzzers that emit a quick beep. The user can utilize sound localization to quickly locate the direction of the beep, and thus the direction of the person who set off the sensor.

Need (c) - Solution

To address need (c), the detection of steps (or obstacles of short to medium height like a low wall), we affixed a GP2Y0A02YK long distance infrared measuring sensor on the crown of the umbrella directly in front of the user. This sensor, during calibration, would come with a tolerance range to adjust for the bobbing of the umbrella during normal walking. Once a step or a low-height obstacle (ie. a short wall) comes within detection range of the sensor, the voltage output of the sensor would be out of the tolerance range, and therefore set off a rapid sequence of vibrations by the vibration panel in the box strapped to the wrist, alerting the user to the step or obstacle.

Need (d) - Solution

In addressing need (d), we implemented wireless capabilities into the RainCane to guide the user to his desired building. A XBEE receiver chip was integrated into the system, and interfaced with a vibration panel in a box strapped to the wrist. Upon approaching the desired building (a XBEE wireless transmitting beacon was used as a substitute in the competition), the buzzer attached to the box will beep faster, alerting the user to his destination.

Need (e) - Solution

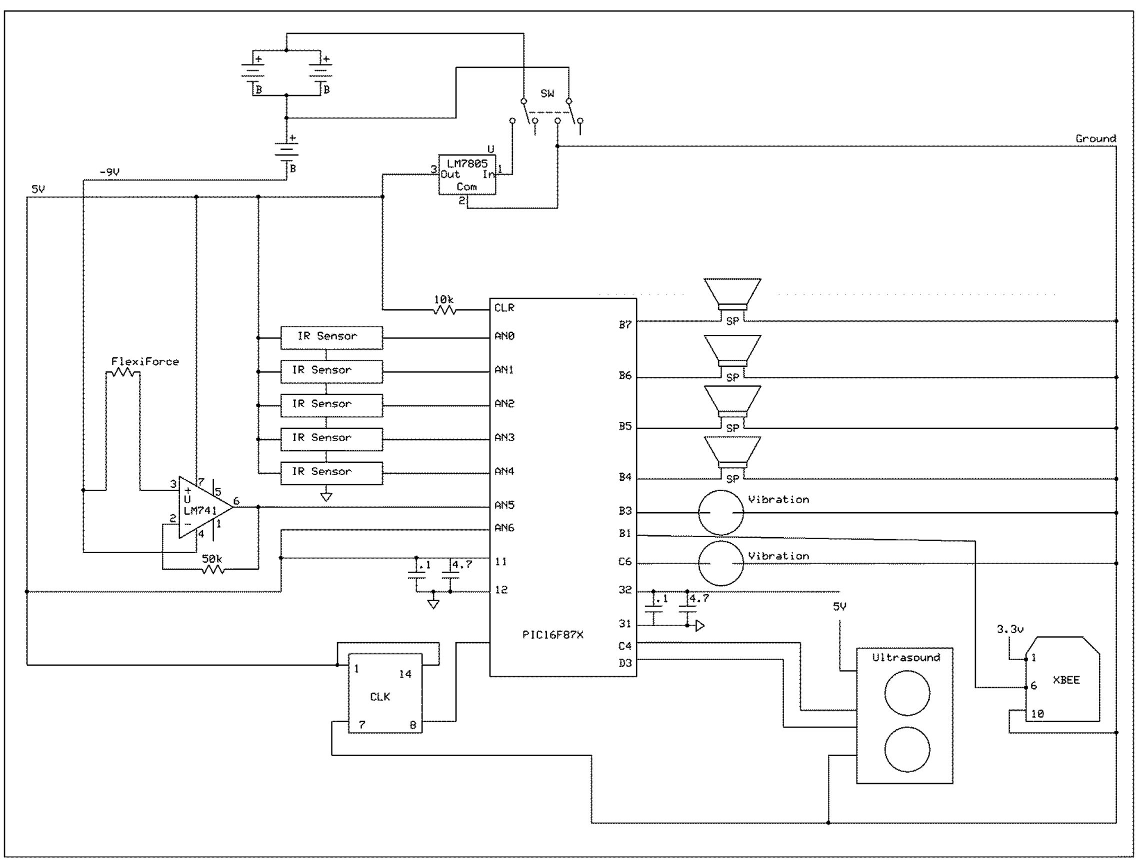
For outdoors use, the device has to be able to endure weather conditions. One difficult problem was to weatherproof a standard cane. We decided to go against convention, and retrofit an umbrella with electronic sensors and feedback devices. The beauty of this solution was that these water-sensitive electronic parts could be sheltered from rain by the crown of the umbrella.

Need (f) – Solution

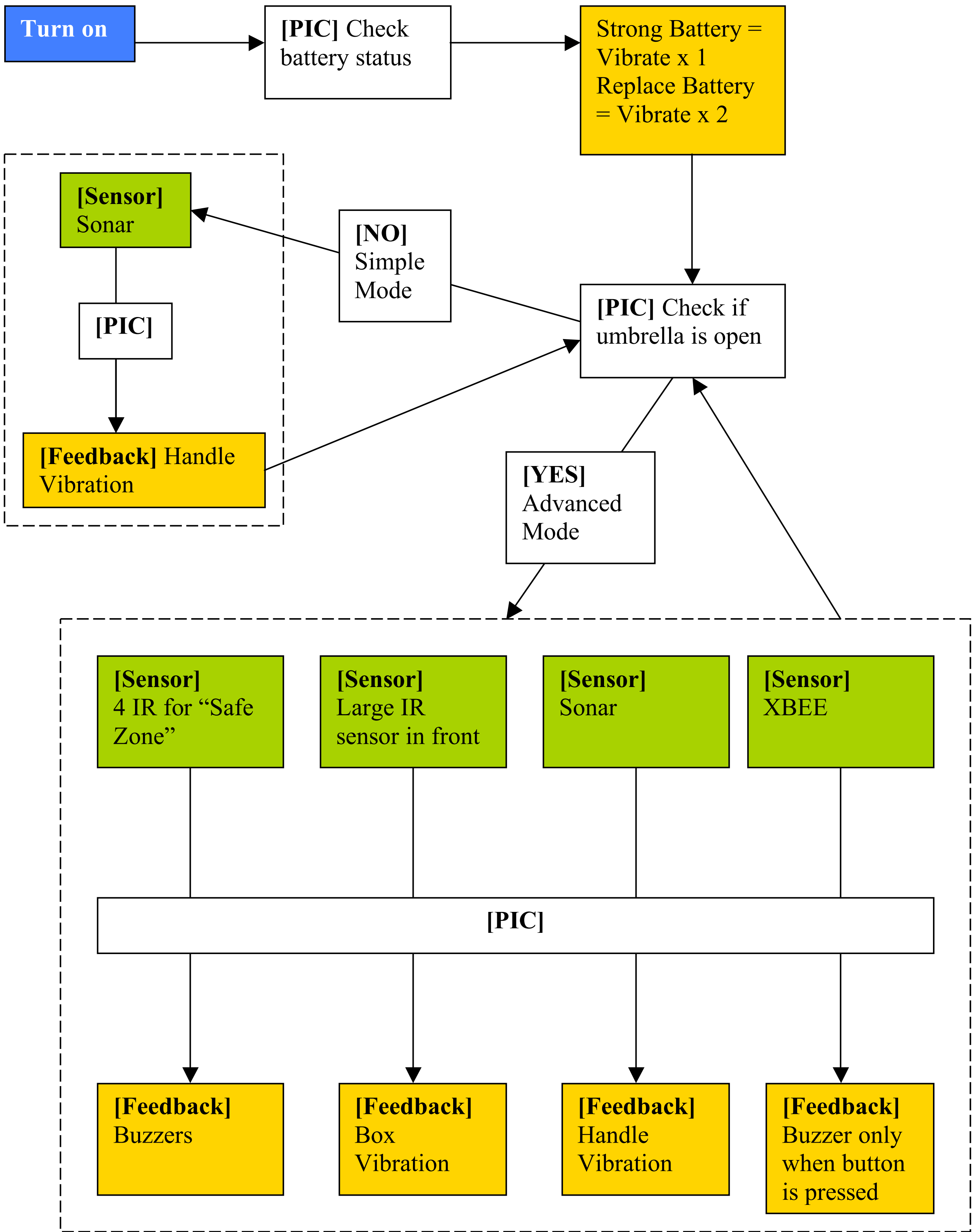
The device needs to double as a cane to be used indoors. The umbrella can be safely closed without any damage to the sensors or feedback devices. Subsequently, the ultrasound sensor on the "hook" of the umbrella can detect obstacles in front, and the extended length of the umbrella allows the user to utilize it as a normal walking stick. This indoors mode is simplistic, but can be further improved by integrating additional sensors to detect more obstacles, such as walls.

Tie-in

In tying together the solutions that address the needs above, we designed a schematic of sensors and feedback devices that are connected together as below:



A simplified version of how the system works is shown below as a block diagram:



Functionality

Idiot-proofing

To improve functionality, we integrated a Flexiforce A201 sensor into the system such that when the umbrella was opened, the sensors would be switched on. Otherwise, if the umbrella was closed, the sensors would not be switched on. This effectively idiot-proofs the system, by ensuring that the user would not accidentally drain the batteries by forgetting to turn the power switch off.

Ultrasound sensor direction

Moreover, we placed a small block of wood on the handle of the umbrella, directly behind the ultrasound sensor, to help the user ensure that the ultrasound sensor would always be facing forward.

Battery status

We implemented a battery status program, so that the user could be alerted to low battery levels and thus change the batteries. If the battery levels are good, the vibration panel would vibrate once when the system is switched on and the umbrella is open. If the battery levels are low and below a certain threshold, the panel would vibrate twice, alerting the user that he needs to replace his batteries soon.

Walking stick

Lastly, we extended the length of the umbrella to allow the user to use it as a walking stick if he closes the umbrella.

Problems Encountered

Insufficient Current

Although our repository of sensors and feedback devices was ideal in building the RainCane, we ran into the problem of having sufficient current for all the devices and sensors in the system. In total, we had 6 sensors, a wireless chip, 2 vibration panels, and 5 high frequency buzzers. To circumvent this problem, we utilized 3 9-volt batteries. Moreover, we had to replace our batteries twice, because of the power-consuming nature of our system.

Difference between breadboard testing and implementation

Our system, when built on the breadboard, and calibrated afterwards, worked perfectly. However, once we soldered everything together and outfitted the umbrella with wires, sensors, and feedback devices, we found that we had to recalibrate all the sensors, which was very time-consuming. This need for recalibration was due to the high level of noise present in the circuit, which was probably due to the long wires we used and the wires being twisted together.

Design Improvements

Printed Circuit Board

We had to deal with many soldering problems; we lost count of the number of times our solders snapped. Even with proper soldering technique, our wires and connections still snapped and we had to spend a lot of time resoldering them. A printed circuit board would effectively remove this problem.

Power

The design could be improved to reduce the power consumption of the system. More efficient devices and sensors could be utilized.

Voice output

Instead of having beeps from buzzers as auditory feedback, a voice system could be utilized. For example, if something activated the sensor located on the left of the user, a micro-speaker could say, “Obstacle on your left”. It would also improve functionality if the micro-speaker could announce the distance from the user to the building based on the pulse width modulation from the wireless chip.

GPS device

Instead of using wireless chips, a GPS device could be integrated into the RainCane. This idea was originally floated, but was discarded due to time constraints.

Indoors mode

To add further functionality to the device, we suggest improving our current simplistic indoors mode. Once the user has gone indoors, he can close the umbrella, thus activating the indoors mode. The ultrasound sensor on the “hook” of the umbrella can act as a detector for obstacles in front, and we suggest implementing two sensors interfaced with vibration panels at the end of the umbrella to detect walls on the left and right of the umbrella.

Portability

Currently, our device is sufficiently portable, but can be further improved such that the box with the circuitry and vibration panel can be further miniaturized.

Conclusion

Our novel design solution to the problem of a smart cane for the blind is unconventional, and has received preliminary positive feedback from a pilot blind user. Furthermore, the device has performed well in use with a time of 9 minutes in the outdoor challenge competition. By utilizing a network of sensors and feedback devices that work in tandem to prove the user with information about his surroundings, he is able to navigate his environment far more effectively than he could with a simple walking stick. With further aforementioned design improvements, and the integration of an indoors mode, the umbrella would be an ideal future replacement for the humble walking stick that millions of blind people today employ.

Appendices

A. Microcontroller ANSI C Code

Appendix A : Microcontroller ANSI C Code

```
#include <16F877.h>
#include <stdio.h>
#include <ctype.h>
#use delay(clock=20000000)
#use rs232(baud=9600, XMIT=TXBEE, RCV = RXBEE, parity=N, bits=8)

// activate front, right buzzer
void frontright_speaker() {
        output_high( PIN_B4 );
        delay_ms(100);
        }

// activate rear, left buzzer
void rearleft_speaker() {
        output_high( PIN_B5 );
        delay_ms(100);
        }

// activate front, left buzzer
void frontleft_speaker() {
        output_high( PIN_B6 );
        delay_ms(100);
        }

// activate rear, right buzzer
void rearright_speaker() {
        output_high( PIN_B7 );
        delay_ms(100);
        }

// activate step vibration panel
void step_sensor1() {
        output_high( PIN_B3 );
        delay_ms(1000);
        }

// activate XBEE buzzer
void xbee1() {
        output_high( PIN_B2 );
        delay_ms(1000);
        output_low( PIN_B2 );
        delay_ms(100);
        }
```

```
// trigger for ultrasound sensor
void trigger() {
	output_high( PIN_D3 );
	delay_us(12);
	output_low( PIN_D3 );
}

void main()
{
	int threshold = 80;//IR threshold
	int threshold1 = 40; //flexiforce threshold
	int threshold2 = 100;//step threshold
	int threshold6 = 5; //ultrasound threshold
	int batterystatus = 230; // battery status voltage threshold
	int value;
	int count;
	int x;
	int i;

	setup_adc(ADC_CLOCK_INTERNAL);
	setup_adc_ports(ALL_ANALOG);
	delay_us(50);

		set_adc_channel(6);
		delay_us(10);
		value = read_adc();

// battery status
		if (value>batterystatus) // if battery status good, 1 beep
		{
			output_high( PIN_B3 );
			delay_ms(2000);
			output_low( PIN_B3 );
			delay_ms(2000);
		}
		if (value<batterystatus) // if battery status bad, 2 beeps
		{
			output_high( PIN_B3 );
			delay_ms(1000);
			output_low( PIN_B3 );
			delay_ms(1000);
			output_high( PIN_B3 );
			delay_ms(1000);
			output_low( PIN_B3 );
			delay_ms(1000);
		}
```

```
//main program

while(TRUE)
          {

// XBEE program - if near XBEE, buzzer
          x = 0;
          for (i=0;i<100;i++) {
                    if (input(PIN_B1)==1) {
                              x = x + 1;
                    }
          delay_us(10);
          }
          for(i=0;i<(x/10);i++){
          output_high(PIN_B2);
          delay_ms(100);
          }
          output_low(PIN_B2);



// ultrasound sensor
          count=0;
          trigger();
          while(!input(PIN_C4));

          while(input(PIN_C4));{
                    delay_us(20);
                    count++;
                    }

// if obstacle detected, vibration, else, nothing
          if(count<=threshold6);
          {
                    output_high(PIN_C6);
          }
          if (count>threshold6);
          {
                    output_low(PIN_C6);
          }



/* read from force sensor to determine if umbrella is open, thus activating the program */
```

```
		set_adc_channel(5);
		delay_us(10);
		value = read_adc();

		if (value>=threshold1)
		{

/* read from frontright sensor - if obstacle detected, beep */
		set_adc_channel(0);
		delay_us(10);
		value = read_adc();

		if (value>=threshold)
		{
			frontright_speaker();
		}
		if (value<threshold)
		{
			output_low(PIN_B4);
			delay_us(10);
		}


/* read from frontleft sensor - if obstacle detected, beep */
		set_adc_channel(1);
		delay_us(10);
		value = read_adc();

		if (value>=threshold)
		{
			frontleft_speaker();
		}
		if (value<threshold)
		{
			output_low(PIN_B6);
			delay_us(10);
		}

/* read from rearright sensor - if obstacle detected, beep */
		set_adc_channel(2);
		delay_us(10);
		value = read_adc();

		if (value>=threshold)
		{
			rearright_speaker();
```

```
		}
		if (value<threshold)
		{
			output_low(PIN_B7);
			delay_us(10);
		}

/* read from rearleft sensor - if obstacle detected, beep */
		set_adc_channel(3);
		delay_us(10);
		value = read_adc();

		if (value>=threshold)
		{
			rearleft_speaker();
		}
		if (value<threshold)
		{
			output_low(PIN_B5);
			delay_us(10);
		}

/* read from step sensor - if step detected, vibrate */
		set_adc_channel(4);
		delay_us(10);
		value = read_adc();

		if (value>threshold2)
		{
			step_sensor1();
		}

		if (value<=threshold2)
		{
			output_low(PIN_B3);
			delay_us(10);
		}

	}
	}
}
```